Doubly Robust Crowdsourcing

Chong Liu   CHONGLIU@CS.UCSB.EDU
Yu-Xiang Wang   YUXIANGW@CS.UCSB.EDU
Department of Computer Science
University of California, Santa Barbara
Santa Barbara, CA 93106, USA

Abstract

Large-scale *labeled* dataset is the indispensable fuel that ignites the AI revolution as we see today. Most such datasets are constructed using crowdsourcing services such as Amazon Mechanical Turk which provides noisy labels from non-experts at a fair price. The sheer size of such datasets mandates that it is only feasible to collect a few labels per data point. We formulate the problem of test-time label aggregation as a statistical estimation problem of inferring the expected voting score. By imitating workers with supervised learners and using them in a doubly robust estimation framework, we prove that the variance of estimation can be substantially reduced, even if the learner is a poor approximation. Synthetic and real-world experiments show that by combining the doubly robust approach with adaptive worker/item selection rules, we often need much lower label cost to achieve nearly the same accuracy as in the ideal world where all workers label all data points.

1. Introduction

The rise of machine learning approaches in artificial intelligence has enabled machines to perform well on many cognitive tasks that were previously thought of as what makes us human. In many specialized tasks, for example, animial recognition in images (He et al., 2015), conversational speech recognition (Xiong et al., 2018), translating Chinese text into English (Hassan et al., 2018), learning-based systems are shown to have reached and even surpassed human-level performances. These remarkable achievements could not have been possible without the many large-scale datasets that are made available by researchers over the past two decades. ImageNet, for instance, has long been regarded as what spawned the AI revolution that we are experiencing today. These labels do not come for free. ImageNet's 11 million images were labeled using Amazon Mechanical Turk (AMT) into more than 15,000 synsets (classes in an ontology). On average, each image required roughly $2-5$ independent human annotations, which were provided by 25,000 AMT workers over a period of three years. We estimate that the cost of getting all these annotations goes well above one million dollars.

As the deep learning models get larger and more powerful every day so as to tackle some of the more challenging AI tasks, their ferocious appetites for even larger labeled dataset have grown tremendously as well. However, unlike the abundant unlabeled data, it is often difficult, expensive, or even impossible to consult expert opinions on large number of items. Here the items can be images, documents, voices, sentences, and so on. Services such as AMT have made it much easier to seek the wisdom of the crowd by having non-experts

(called workers in the remainder of this paper) to provide many noisy annotations at a much lower cost. A large body of work has been devoted to finding more scalable solutions. These include a variety of label-aggregation methods (Sheng et al., 2008; Welinder et al., 2010; Zhang et al., 2016; Zhou et al., 2015), end-to-end human-in-the-loop learning (Khetan et al., 2018), online/adaptive worker selections (Branson et al., 2017; Van Horn et al., 2018) and so on. At the heart of these approaches, there are various ways to evaluate individual worker performances and quantify the uncertainty in their provided labels.

In this paper, we take a pre-trained crowdsourcing model with worker evaluation as a blackbox and consider the problem of true label inference for new data points. We formulate this problem as a statistical estimation problem and propose a number of ways to radically reduce the number of worker annotations.

1. **Worker imitation** We propose to imitate each worker with a simple supervised learner that learns to predict the worker's label using the item feature.

2. **Doubly robust crowdsourcing (DRC)** By tapping into the literature on doubly robust estimation, we design algorithms that exploit the possibly unreliable imitation agents and significantly reduce the estimation variance (hence annotation cost) while remaining unbiased.

3. **Adaptive worker/item selection (AWS/AIS)** We propose to bootstrap the imitation agents' confidence estimates to adaptively filter out high confidence items and select the most qualified workers for low-confidence item, without additional cost.

Our results are summarized as follows.

1. We theoretically show that DRC technique can be used to generically improve any given crowdsourcing models using any nontrivial learned imitation agents.

2. Synthetic and real-world experiments show DRC improves the label accuracy over the standard probabilistic inference with Dawid-Skene and majority voting models in almost all budget levels and all datasets.

3. AWS and AIS often reduce the cost by orders of magnitudes, while enjoying the same level of accuracy. On several datasets, the proposed technique can often get away with much fewer annotations per item while achieving almost the same accuracy that can be obtained by having all workers annotating all items.

2. Related Work

In this section, we briefly summarize the related work.

Our study is motivated by the many trailblazing approaches in label-aggregation including the wisdom-of-crowds (Welinder et al., 2010), Dawid-Skene model (Dawid and Skene, 1979; Zhang et al., 2016), minimax entropy approach (Zhou et al., 2015), permutation-based model (Shah et al., 2020), worker cluster model (Imamura et al., 2018), crowdsourced regression model (Ok et al., 2019) and so on. Our contribution is complementary as we can take any of these models as blackboxes and hopefully improve their true-label inference.

Doubly robust techniques originates from the causal inference literature (Rotnitzky and Robins, 1995; Bang and Robins, 2005) and the use of it for variance reduction had led to several breakthroughs in machine learning (e.g., Johnson and Zhang, 2013; Wang et al., 2013). We drew our inspirations directly from the use of doubly robust techniques in the off-policy evaluation problem in bandits and reinforcement learning (Dudík et al., 2014; Jiang and Li, 2016; Wang et al., 2017). The variance analysis and weight-clipping are adapted from the calculations in Dudík et al. (2014) and Wang et al. (2017) with some minor differences. To the best of our knowledge, this is the first paper considering doubly robust techniques in crowdsourcing.

Our idea of adaptive item/worker selection is inspired by the recent work of Branson et al. (2017) and Van Horn et al. (2018). They propose an AI-aided approach that reduces the number of worker labels per item to be smaller than 1 in an object detection task. The key idea is to train a computer vision algorithm to detect the bounding boxes using the aggregated labels that have been obtained thus far and if the algorithm achieves a high confidence on a new image, then the annotation provided by the algorithm is taken.

The differences of our work is twofold. First, our use of supervised learner is not to predict the true labels but rather to imitate workers. Second, our confidence measure is determined by supervised learners' approximation to what all workers would say about an item, rather than as a prior distribution added to model-based probabilistic inference.

3. Problem Setup

In this section, we introduce the notations and formulate the problem as a statistical estimation problem.

3.1 Notations

Suppose we have n items, m workers, and k classes. We adopt the notation $[k] := \{1, 2, 3, ..., k\}$. Each item $j \in [n]$ is described as a d-dimensional feature vector x_j, and the feature matrix is $X = [x_1, x_2, \cdots, x_n]^\top \in \mathbb{R}^{n \times d}$. Each item $j \in [n]$ also has a hidden true label $y_j \in [k]$ which indicates the correct class that item j belongs to.

Workers, such as those on AMT, are requested to classify items into one of the k classes. We denote the label that worker $i \in [m]$ assigns to item j as $\ell_{ij} \in [k]$. It is important to distinguish the worker-produced labels ℓ_{ij} with the true label y_j, as the workers are considered non-experts and they make mistakes. From here onwards, we will refer to the potentially noisy and erroneous labels from workers as "annotations". Conveniently, we also collect ℓ_{ij} into a matrix $L \in ([k] \cup \{\perp\})^{m \times n}$, where any entries in L that are $\perp$ are unobserved labels. We use $\Omega \subset [m] \times [n], \Omega_i \subset [n], \Omega_j \subset [m]$ to denote the indices of the observed annotations, indices of all items worker i annotated and indices of all workers that annotated item j respectively. For a generic item (x, y), Ω_x collects the indices of workers who annotated the item and the corresponding annotation is denoted by ℓ_i for each $i \in \Omega_x$.

3.2 Problem Statement

The goal of the paper is related to but different from the standard crowdsourcing problem which aims at learning a model that one can use to infer the true label $y_1, ..., y_n$ using noisy

annotations $L[\Omega]$ (and sometimes item features X). Many highly practical models were proposed for that task already (Dawid and Skene, 1979; Welinder et al., 2010; Zhang et al., 2016; Zhou et al., 2015; Shah et al., 2020).

Complementary to the existing work that mainly focuses on label inference, we consider the problem of cost-saving. Specifically, we would like to design algorithms to reduce the expected number of new annotations needed to label a new item. The algorithm uses a pre-trained crowdsourcing model as well as the training dataset X and $L[\Omega]$.

3.3 Dawid-Skene Model and Score Functions

The primary model that we work with in this paper is the Dawid-Skene (DS) model (Dawid and Skene, 1979; Zhang et al., 2016), which assumes the following data generating process.

1. For each $j \in [n]$, $y_j \sim \text{Categorical}(\tau)$.

2. For each $j \in [n], i \in [m]$, $\ell_{ij} \sim \text{Categorical}(\mu_{y_j,i})$.

3. We observe ℓ_{ij} with probability π_{ij}.

where τ and $\mu_{y,i}$ denote the probability distributions defined on $[k]$. In particular, $\mu_{y,i}$ is the column y of the *confusion matrix* of worker i, which the DS model uses to describe $\mathbb{P}_i(\ell|y)$. We denote the confusion matrix associated with worker i by $\mu_i \in \mathbb{R}^{k \times k}$. Once the DS model is learned, we can make use of the learned parameters τ and μ to infer the true labels using worker annotations via the posterior belief

$$P(y|\ell_1, \ell_2, ..., \ell_m) \propto P(y) \prod_{i=1}^{m} P(\ell_i|y) = \tau[y] \prod_{i=1}^{m} \mu_i[\ell_i, y]. \tag{1}$$

Take log for both sides and dropping the additive constant, we obtain the *score function* that is induced by the DS model

$$S_{\text{DS}}(y|\ell_i \ \forall i \in \Omega_x) = \log \tau[y] + \sum_{i=1}^{m} \log \mu_i[\ell_i, y]. \tag{2}$$

This is a weighted voting rule based on a pre-trained DS model. Similarly, we can cast the inference procedure of other crowdsourcing models as maximizing such a score function as well. For example, in the Majority Voting (MV) approach,

$$S_{\text{MV}}(y|\Omega_x) = \sum_{i=1}^{m} \mathbf{1}(\ell_i(x) = y), \tag{3}$$

where $\mathbf{1}(\cdot)$ is the indicator function. Notably, no training datasets are needed for majority voting. The exposition above suggests that test time involves collecting a handful of worker annotations (choosing Ω_x) and calculating a voting score specified by the crowdsourcing model in a form of

$$S(y|\Omega_x) = \sum_{i \in \Omega_x} S_i(y, \ell_i(x)). \tag{4}$$

where S_i is supplied by the model that connects annotation ℓ_i to label y. Then the label y that maximizes the score is chosen.

3.4 A Statistical Estimation Framework

In the ideal world, when money is not a concern, we will poll *all* workers and calculate

$$S(y|[m]) = \sum_{i=1}^{m} S_i(y, \ell_i(x)). \tag{5}$$

In practice, however, just as we cannot afford to poll all voters to estimate who is winning the presidential election, we cannot afford to poll everyone to annotate a single data point either. But do we have to?

Notice that we can frame the question as a classical point estimation problem in statistics, where the statistical quantity of interest is

$$v^x(y) := \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m} S_i(y, \ell_i(x))\right], \tag{6}$$

the expectation of the ideal world score function (5), rescaled by $1/m$. In the above, the expectation is taken over the randomness in worker's annotation. For example, if we select each worker independently with probability π, then the approach used in (2) and (3) would be an unbiased estimate of $v^x(y)$, if we rescale them by a factor of π^{-1}.

The advantage of translating the problem into a classical statistical estimation problem is that there is now a century of associated literature that we can tap into, including those on adaptive sampling and variance reduction techniques. We emphasize that while we will be using a crowdsourcing model, for example, the Dawid-Skene model, we do not assume that the data is generated according to the model. In fact, we are not imposing any restrictions on how workers annotate items, except that

1. $\ell_i(x) \; \forall i \in [m]$ are mutually independent given any item x.

2. $\text{Var}[S_i(y, \ell_i(x))] < +\infty \quad \forall x, y$.

These are very mild assumptions that are typically true in practice. It is generally difficult to analytically model human behaviors because it depends on how the item is presented to worker as well as the worker's knowledge and cognitive processes. The agnostic learning point of view helps disentangle the approximation-theoretic questions from the statistical question of estimating the best approximation possible using a given crowdsourcing model.

The remainder of the paper will be about designing estimators of $v^x(y)$ that achieves accurate label inference at a low cost and their corresponding theory and experiments. To avoid any confusions, we emphasize again that item x is fixed. All the estimators are defined for each y separately. $\ell_1, ..., \ell_m$ are random variables that comes out of the unknown process of workers looking at the item x. Whenever the dependence is clear from context, we drop the conditioning on x for better readability.

4. Benchmark Approaches

In this section, we describe a few baseline approaches for estimating $v^x(y)$ and their corresponding cost in number of annotations.

4.1 Ideal World (IW) Estimator

In the ideal world, all workers are required to label x:

$$\hat{v}_{\text{IW}}(y) = \frac{1}{m} \sum_{i=1}^{m} S_i(y, \ell_i). \tag{7}$$

This estimator incurs cost of m and it is unbiased, with variance of $\frac{1}{m^2} \sum_{i=1}^{m} \text{Var}[S_i(y, \ell_i(x))]$. This is arguably the best one can do with additional information.

4.2 Importance Sampling (IS) Estimator

A more affordable approach is to directly sample the workers. Specifically, we will include worker i independently with probability π_i [1].

$$\hat{v}_{\text{IS}}(y) = \frac{1}{m} \sum_{i=1}^{m} \frac{\mathbf{1}(i \in \Omega)}{\pi_i} S_i(y, \ell_i). \tag{8}$$

The expected cost of the IS estimator is $\sum_{i \in [m]} \pi_i$ and it is clearly an unbiased estimator.

Theorem 1. *The $\hat{v}_{\text{IS}}(y)$ is unbiased and*

$$\text{Var}[\hat{v}_{\text{IS}}(y)] = \frac{1}{m^2} \left(\sum_{i=1}^{m} \frac{1}{\pi_i} \text{Var}[S_i(y, \ell_i)] + (\frac{1}{\pi_i} - 1)\mathbb{E}[S_i(y, \ell_i)]^2 \right).$$

Proof. By the independence of sampling,

$$\mathbb{E}[\hat{v}_{\text{IS}}(y)] = \frac{1}{m} \sum_{i=1}^{m} \mathbb{E}\Big[\mathbf{1}(i \in \Omega)\frac{1}{\pi_i}S_i(y, \ell_i) \Big]$$

$$= \frac{1}{m} \sum_{i=1}^{m} \pi_i \frac{1}{\pi_i}\mathbb{E}[S_i(y, \ell_i)]$$

$$= v^x(y).$$

To calculate the variance, we use the independence and then apply the law of total variance on each i:

$$\text{Var}[\hat{v}_{\text{IS}}(y)] = \frac{1}{m^2} \sum_{i=1}^{m} \frac{1}{\pi_i^2} \text{Var}[\mathbf{1}(i \in \Omega)S_i(y, \ell_i)]$$

$$= \frac{1}{m^2} \sum_{i=1}^{m} \frac{1}{\pi_i^2} \Big(\mathbb{E}[(\mathbf{1}(i \in \Omega))^2]\mathbb{E}[(S_i(y, \ell_i))^2] - \mathbb{E}[\mathbf{1}(i \in \Omega)]^2\mathbb{E}[S_i(y, \ell_i)]^2 \Big)$$

$$= \frac{1}{m^2} \sum_{i=1}^{m} \frac{1}{\pi_i^2} \Big(\pi_i\text{Var}[S_i(y, \ell_i)] + \pi_i\mathbb{E}[S_i(y, \ell_i)]^2 - \pi_i^2\mathbb{E}[S_i(y, \ell_i)]^2 \Big)$$

$$= \frac{1}{m^2} \left(\sum_{i=1}^{m} \frac{1}{\pi_i} \text{Var}[S_i(y, \ell_i)] + (\frac{1}{\pi_i} - 1)\mathbb{E}[S_i(y, \ell_i)]^2 \right).$$

$\square$

---

1. This is called a Poisson sampling (Särndal et al., 2003) in the survey sampling theory.

Remark. If $\pi_i \equiv \pi$, then we are essentially doing the standard probabilistic inference as in (2) and (3). When $\pi_i = 1$, IS trivially subsumes IW (7) as a special case. Moreover, since x is fixed, the sampling π_i can be chosen as a function of the item x without affecting the above results.

4.3 Direct Method (DM)

Finally, there is an option that comes with no cost. Recall that we have a dataset X and L that were used to train the crowdsourcing model at our disposal. We can reuse the dataset and train m supervised learners to imitate each worker's behavior. Let $\hat{\ell}_1, ..., \hat{\ell}_m$ be the fictitious annotations provided by these supervised learners, we can simply plug them into the ideal world estimator (7) without any cost,

$$\hat{v}_{\text{DM}}(y) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{E}[S_i(y, \hat{\ell}_i)]. \tag{9}$$

Following the convention in the contextual bandits literature (Jiang and Li, 2016), we call this approach the direct method. The additional $\mathbb{E}$ is introduced to capture the case when supervised learner outputs a soft annotation $\hat{\ell}_i$.

The variance of this approach is 0. However, as we mentioned previously, we can never hope to faithfully learn human behaviors, especially when we only have a small number of annotations in the training data for each worker i. As a result, (9) may suffer from a bias that does not vanish even as $m \to \infty$.

5. Main Results

In this section, we adapt an old statistical technique, doubly robust estimation, to crowdsourcing problem.

5.1 Doubly Robust Crowdsourcing

As we established in the last section, IS estimator is unbiased but suffers from a large variance, especially when we would like to cut cost and use a small sampling probability. The DM estimator incurs no additional annotation cost and has no variance, but it can potentially suffer from a large bias due to supervised learners not imitating the workers well enough.

Doubly robust estimation (Rotnitzky and Robins, 1995; Dudík et al., 2014) is a powerful technique that allows us to reduce the variance using a DM estimator while retaining the unbiasedness, hence getting the best of both worlds. The doubly robust estimator works as follows:

$$\hat{v}_{\text{DR}}(y) = \frac{1}{m} \sum_{i=1}^{m} \left(\mathbb{E}[S_i(y, \hat{\ell}_i)] + \frac{\mathbf{1}(i \in \Omega)}{\pi_i} (S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]) \right). \tag{10}$$

The doubly robust estimator can be thought of using the DM as a baseline and then use IS to estimate and correct the bias. Provided that the supervised learners are able to provide a nontrivial approximation of the workers, the doubly robust estimator is expected to reduce

the variance. Just to give two explicit examples of $\hat{v}_{DR}(y)$, under the Dawid-Skene model, the doubly robust estimator is

$$\frac{1}{m}\sum_{i=1}^{m}\log P_{\mu_i}(\hat{\ell}_i|y) + \frac{\mathbf{1}(i \in \Omega)}{\pi_i}\log\frac{P_{\mu_i}(\ell_i|y)}{P_{\mu_i}(\hat{\ell}_i|y)}. \tag{11}$$

Similarly, for the majority voting model, we can write

$$\frac{1}{m}\sum_{i=1}^{m}e_{\hat{\ell}_i} + \frac{1}{\pi_i}\mathbf{1}(i \in \Omega)(e_{\ell_i} - e_{\hat{\ell}_i}), \tag{12}$$

where e_ℓ is the basis vector where ℓ indicates the location of 1, otherwise 0.

Theorem 2 (DRC). *The doubly robust estimator* (10) *is unbiased and its variance is:*

$$\frac{1}{m^2}\left(\sum_{i=1}^{m}\frac{1}{\pi_i}\mathrm{Var}[S_i(y, \ell_i)] + \left(\frac{1}{\pi_i} - 1\right)\mathbb{E}[S_i(y, \ell_i) - S_i(y, \hat{\ell}_i)]^2\right).$$

Proof Sketch. Note that the first part $\frac{1}{m}\sum_{i=1}^{m}\mathbb{E}[S_i(y, \hat{\ell}_i)]$ of the estimator is not random. The result follows directly by invoking Theorem 1 on the second part of the estimator, which is an importance sampling estimator of the bias. $\square$

Proof. By the independence of sampling,

$$\mathbb{E}[\hat{v}_{\mathrm{DR}}(y)] = \frac{1}{m}\sum_{i=1}^{m}\mathbb{E}\big[\mathbb{E}[S_i(y, \hat{\ell}_i)]\big] + \frac{1}{\pi_i}\mathbb{E}\big[\mathbf{1}(i \in \Omega)(S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)])\big]$$

$$= \frac{1}{m}\sum_{i=1}^{m}\mathbb{E}\big[\mathbb{E}[S_i(y, \hat{\ell}_i)]\big] + \pi_i\frac{1}{\pi_i}\mathbb{E}[S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]]$$

$$= \frac{1}{m}\sum_{i=1}^{m}\mathbb{E}[S_i(y, \ell_i)]$$

$$= v^x(y).$$

To calculate the variance, we use the independence and then apply the law of total variance on each i:

$$\mathrm{Var}[\hat{v}_{\mathrm{DR}}(y)] = \frac{1}{m^2}\sum_{i=1}^{m}\frac{1}{\pi_i^2}\mathrm{Var}\big[\mathbf{1}(i \in \Omega)(S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)])\big]$$

$$= \frac{1}{m^2}\sum_{i=1}^{m}\frac{1}{\pi_i^2}\Big(\mathbb{E}[(\mathbf{1}(i \in \Omega))^2]\mathbb{E}[(S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)])^2]$$

$$\qquad - \mathbb{E}[\mathbf{1}(i \in \Omega)]^2\mathbb{E}[S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]]^2\Big)$$

$$= \frac{1}{m^2}\sum_{i=1}^{m}\frac{1}{\pi_i^2}\Big(\pi_i\mathrm{Var}[S_i(y, \ell_i)] + \pi_i\mathbb{E}[S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]]^2$$

$$\qquad - \pi_i^2\mathbb{E}[S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]]^2\Big)$$

$$= \frac{1}{m^2}\left(\sum_{i=1}^{m}\frac{1}{\pi_i}\mathrm{Var}[S_i(y, \ell_i)] + \left(\frac{1}{\pi_i} - 1\right)\mathbb{E}[S_i(y, \ell_i) - S_i(y, \hat{\ell}_i)]^2\right).$$

$\square$

Remark. First, if workers are deterministic, the first part of the variance $\mathrm{Var}[S_i(y, \ell_i)] \equiv 0$. Second, if the supervised learner imitates workers perfectly *in expectation*, the second part of the variance vanishes. Finally and most importantly, the supervised learner does *not* have to be perfect. In the simple case of a deterministic workers, the percentage of agreements between supervised learners and their human counterparts directly translate into a reduction of the variance of about the same percentage, for free.

The third point is especially remarkable as it implies that even a trivial surrogate that outputs a label at random could lead to a $1/k$ factor reduction of the variance. In addition, a good set of worker imitators with 90% accuracy can lead to an order of magnitude smaller variance and hence allow us to incur a much lower cost on average. We will illustrate the effects of doubly robust estimation more extensively in the experiments. This feature ensures that our proposed method remains applicable even in the case when the training dataset contain few annotations from some subset of the features.

5.2 Confidence-Based Adaptive Sampling

Doubly robust estimation allows us to reduce the variance. However, doubly robust is still an importance sampling-based method that requires the number of new annotations to be at least linear in the number of data points to label. In this section, we propose using supervised worker imitation to obtain confidence estimates for free and using them to construct confidence-based adaptive sampling schemes.

We propose two rules.

1. **Adaptive item selection** For each new data point, run DM first. If DM predicts label y with an overwhelming confidence, then chances are, there is no need to collect more annotations. If not, human workers are needed.

2. **Adaptive worker selection** We can adaptively choose which worker to annotate a given item. Instead of sampling at random with probability π, we choose a set of adaptive sampling probability $\pi_1, \cdots, \pi_m$ that makes high confidence workers more likely to be selected. As different workers have different skill sets, confidence may depend strongly on each item x. We propose to calculate such item-dependent confidence using outcome of the imitated workers and the confusion matrices from the DS model.

In both cases, we need a way to measure confidence given a probability distribution. A threshold is introduced to decide whether accept predicted labels or not (Branson et al., 2017). Margin in multi-class classification is defined as the difference between the score of true label and the largest score of other labels (Mohri et al., 2012). Inspired by them, we define confidence margin of a probability as follows.

Definition 1 (Confidence Margin). *Given a discrete probability distribution $\pi_1, ..., \pi_m$, its confidence margin ρ is defined as the difference between the largest probability and the second largest one.*

Based on confidence margin, we propose three new methods: DRC with Adaptive Item Selection (DRC-AIS), DRC with Adaptive Worker Selection (DRC-AWS), and the combination DRC-AWS-AIS. In DRC-AIS, DM is performed on all labels. For each item, the surrogate label given by DM follows (1) to get the posterior belief, which describes how confidently DM gives the label of this item. Based on posterior belief, its confidence margin ρ_{AIS} is compared with the given confidence margin parameter ρ. If ρ_{AIS} is larger, DRC-AIS takes the surrogate label provided by DM with no worker cost, otherwise, DRC-AIS follows the regular DRC model which incures cost. In DRC-AWS, again DM runs first and gets surrogate labels $\hat{\ell}_i$. Then from each worker i's confusion matrix we can get the labeling probability $P(\hat{\ell}_i|y)$, whose confidence margin is used as the worker score γ_i and $\gamma_1, \cdots, \gamma_m$ are normalized to be a distribution. For each item, worker i will be sampled with probability γ_i. However, in this case the sampling probability for each worker is usually very small, thus, we can introduce a parameter λ to multiply with γ_i to increase the sampling probability. If a worker is sampled, the corresponding label is used as regular DRC model.

Table 1 shows the summary of benchmark and DRC approaches. As we can see, IW and IS take the fewest input elements while IW uses the most worker cost. DM uses no cost because it only takes advantage of surrogate labels given by worker imitation. DRC has similar cost as IS while it is expected to improve the ground truth inference with less variance. Thanks to the confidence-based adaptive sampling techniques, DRC-AIS and DRC-AWS are able to save more cost than DRC.

Table 1: Summary of benchmark and DRC approaches in DS models. L, X are annotation matrix and item feature matrix. μ, τ are probability distributions from DS models. f denotes the supervised classifier for worker imitation.

Methods	Input	Sampling	Cost
IW	L	No	$O(nm)$
IS	L	π	$O(\pi nm)$
DM	$X + f$	No	0
DRC	$L + \mu + \tau(y) + f$	π	$O(\pi nm)$
DRC-AIS	$L + \mu + \tau(y) + f$	π	$O(\pi n^* m)$
DRC-AWS	$L + \mu + \tau(y) + f$	$\pi_{1:m}$	$O(\sum_{i \in [m]} \pi_i n)$

$n^* \ll n$ denotes number of items AIS selected as low-confidence.

5.3 Weight-Clipping in DRC

Adaptive worker selection involves making selection probability π larger for some workers while smaller for others. According to Theorem 2, the variance is proportional to $\sum_i \pi_i^{-1}$, hence even a single π_i being close to 0 would result in a huge variance. In off-policy evaluation (Wang et al., 2017) problems this issue is addressed by clipping the importance

weight at a fixed threshold η. This results in the clipped doubly robust estimator.

$$\hat{v}_{\mathrm{DR}_\eta}(y) = \frac{1}{m} \sum_{i=1}^{m} \left(\mathbb{E}[S_i(y, \hat{\ell}_i)] + \mathbf{1}(i \in \Omega) \min\{\eta, \pi_i^{-1}\} (S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]) \right). \qquad (13)$$

Its bias and variance are given as follows.

Theorem 3. *The clipped doubly robust estimator obeys that:*

$$\mathrm{Bias}(\hat{v}_{\mathrm{DR}_\eta}(y)) = \left| \frac{1}{m} \sum_{i=1}^{m} \min\{\pi_i \eta - 1, 0\} \mathbb{E}[S_i(y, \ell_i) - S_i(y, \hat{\ell}_i)] \right|.$$

$$\mathrm{Var}[\hat{v}_{\mathrm{DR}_\eta}(y)] = \frac{1}{m^2} \sum_{i=1}^{m} \min\{\eta^2 \pi_i^2, 1\} \left(\frac{1}{\pi_i} \mathrm{Var}[S_i(y, \ell_i)] + \left(\frac{1}{\pi_i} - 1 \right) \mathbb{E}[S_i(y, \ell_i) - S_i(y, \hat{\ell}_i)]^2 \right).$$

Proof. By definition of bias,

$$\mathrm{Bias}(\hat{v}_{\mathrm{DR}_\eta}(y)) = \left| \mathbb{E}[\hat{v}_{\mathrm{DR}_\eta}(y)] - \frac{1}{m} \sum_{i=1}^{m} \mathbb{E}[S_i(y, \ell_i)] \right|$$

$$= \left| \frac{1}{m} \sum_{i=1}^{m} \left(\mathbb{E}[\mathbb{E}[S_i(y, \hat{\ell}_1)]] + \mathbb{E}\left[\mathbf{1}(i \in \Omega) \min\{\eta, \pi_i^{-1}\} (S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]) \right] \right. \right.$$

$$\left. \left. - \mathbb{E}[S_i(y, \ell_i)] \right) \right|$$

$$= \left| \frac{1}{m} \sum_{i=1}^{m} \left(\mathbb{E}[S_i(y, \hat{\ell}_1)] + \min\{\pi_i \eta, 1\} \mathbb{E}[S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]] - \mathbb{E}[S_i(y, \ell_i)] \right) \right|$$

$$= \left| \frac{1}{m} \sum_{i=1}^{m} \min\{\pi_i \eta - 1, 0\} \mathbb{E}[S_i(y, \ell_i) - S_i(y, \hat{\ell}_i)] \right|.$$

And the variance can be calculated as,

$$\mathrm{Var}[\hat{v}_{\mathrm{DR}_\eta}(y)] = \frac{1}{m^2} \sum_{i=1}^{m} \min\{\eta^2, \pi_i^{-2}\} \mathrm{Var}[\mathbf{1}(i \in \Omega)(S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)])]$$

$$= \frac{1}{m^2} \sum_{i=1}^{m} \min\{\eta^2, \pi_i^{-2}\} \left(\mathbb{E}[(\mathbf{1}(i \in \Omega))^2] \mathbb{E}[(S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)])^2] \right.$$

$$\left. - \mathbb{E}[\mathbf{1}(i \in \Omega)]^2 \mathbb{E}[S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]]^2 \right)$$

$$= \frac{1}{m^2} \sum_{i=1}^{m} \min\{\eta^2, \pi_i^{-2}\} \left(\pi_i \mathrm{Var}[S_i(y, \ell_i)] + \pi_i \mathbb{E}[S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]]^2 \right.$$

$$\left. - \pi_i^2 \mathbb{E}[S_i(y, \ell_i) - \mathbb{E}[S_i(y, \hat{\ell}_i)]]^2 \right)$$

$$= \frac{1}{m^2} \sum_{i=1}^{m} \min\{\eta^2 \pi_i^2, 1\} \left(\frac{1}{\pi_i} \mathrm{Var}[S_i(y, \ell_i)] + (\frac{1}{\pi_i} - 1) \mathbb{E}[S_i(y, \ell_i) - S_i(y, \hat{\ell}_i)]^2 \right).$$

$\square$

219

Remark. The bias bound indicates that only those workers we clipped who contribute to the bias. The variance bound implies that the part of variance from Worker i is reduced to $O(\min\{1/\pi_i, \eta^2\pi_i\})$ from $O(1/\pi_i)$. If the total amount of additional Bias^2 introduced by the clipping is smaller than the corresponding savings in the variance, then clipping makes the estimator more accurate in Mean Squared Error (MSE). The theory inspires us to design an algorithm to automatically choose the threshold.

Remark (Automatic choice of threshold). Assume $\ell_1, ..., \ell_m$ are deterministic, $|\mathbb{E}[S_i(y, \ell_i) - S_i(y, \hat{\ell}_i)]| \leq \epsilon$. Then the bias of $\hat{v}_{\text{DR}_\eta}(y)$ can be bounded by $\epsilon \sum_i \mathbf{1}(\pi_i^{-1} > \eta)/m$, and variance of $\hat{v}_{\text{DR}_\eta}(y)$ can be bounded by $\epsilon^2\eta^2/m$. Recall that MSE can be decomposed in to Bias^2 + Var. The optimal choice of η that minimizes this upper bound is the one that minimizes $|\eta - \sum_i \mathbf{1}(\pi_i^{-1} > \eta)/\sqrt{m}|$. This can be found numerically in time $O(m\log(m))$ by sorting $[\pi_1^{-1}, ..., \pi_m^{-1}]$ and applying binary search.

6. Experiments

In this section, we report our experimental results, including both synthetic and real-world experiments.

6.1 Synthetic Experiments

First we describe our experimental settings and then move to synthetic experiment results.

6.1.1 EXPERIMENTAL SETTINGS

We are using supervised classification datasets to do synthetic experiments. Due to absence of labeling matrix, we follow the workflow shown in Figure 1 to do worker imitation to generate crowdsourcing datasets, which has three steps.

1. We starts with the raw dataset in step 1. If the dataset was split into training and test parts, we combine them together.

2. In step 2, we uniformly sample the dataset into two equal parts, one for training and the other for test. In order to remove the randomness of this splitting process, the sampling index is fixed and saved for all further experiments. Training means we use this part of data to train m decision trees to simulate the generating process of crowdsourcing labels. For one item only $\sqrt{d}$ features, a subset of all d features, can be observed by each tree where d is the total number of features. Then these m decision trees are used to make predictions on the test set to obtain the item label matrix.

3. In step 3, we uniformly sample the test part of step 2 into two equal parts, one working as source part and the other working as target part. For all experiments, this sampling process will be repeated 20 times. Also, m classifiers, one for each worker, are trained on source dataset to simulate worker behaviors to give surrogate labels for all DRC approaches. For all synthetic experiments, $m = 50$. Evaluations are performed on the target label matrix.

We use five classification datasets, Segment, Satimage, Usps (Hull, 1994), Pendigits, and Mnist (LeCun et al., 1998), collected by Libsvm (Chang and Lin, 2011), which are all
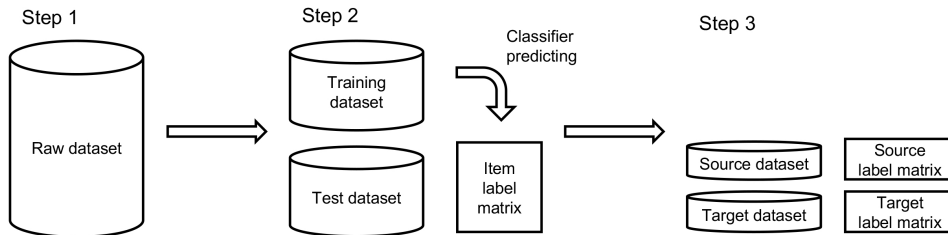
Figure 1: Workflow of generating crowdsourcing datasets with item features.

publicly available. Table 2 shows their statistics of test dataset and item label matrix of Step 2.

Table 2: Statistics of synthetic datasets.

Dataset	# item	# worker	# dimension	# class
Segment	$1,155$	50	19	7
Satimage	$3,217$	50	36	6
Usps	$4,649$	50	256	10
Pendigits	$5,496$	50	16	10
Mnist	$35,000$	50	780	10

All experimental results, in figures or tables, are presented after repeating 20 times with 98% asymptotic confidence interval of the expected accuracy based on inverting Wald's test, that is,

$$\mu \pm 1.96\sigma/\sqrt{20}, \tag{14}$$

where μ, σ are the mean and stand error of accuracy. Based on Wald's test, statistical conclusions can be made with 98% confidence.

6.1.2 Algorithm Comparison

In order to show our approach DRC is able to infer true labels with low worker cost, we compare our DRC approach with Importance Sampling (IS). In particular, we do experiments with both Dawid-Skene (DS) model and Majority Voting (MV) model. Therefore, we are comparing DRC-DS, DS, DRC-MV, and MV. Moreover, we have Direct Method (DM) and Classifiers trained on Inferred Labels (CI) as baselines. CI means classifiers are trained with inferred labels from source part and then make predictions for the target part, thus incurring no labeling cost. In detail, we do Poisson sampling over workers with π going from 0.1 to 1.0 with interval of 0.1. Decision trees with maximum depth 3 are used to generate surrogate labels for DRC-DS, DRC-MV, and DM.

Results are shown in Figure 2, which have three observations.

- Given the same worker sampling rate, DRC-DS outperforms DS and DRC-MV works better than MV on all datasets, which shows the effectiveness of DRC and matches
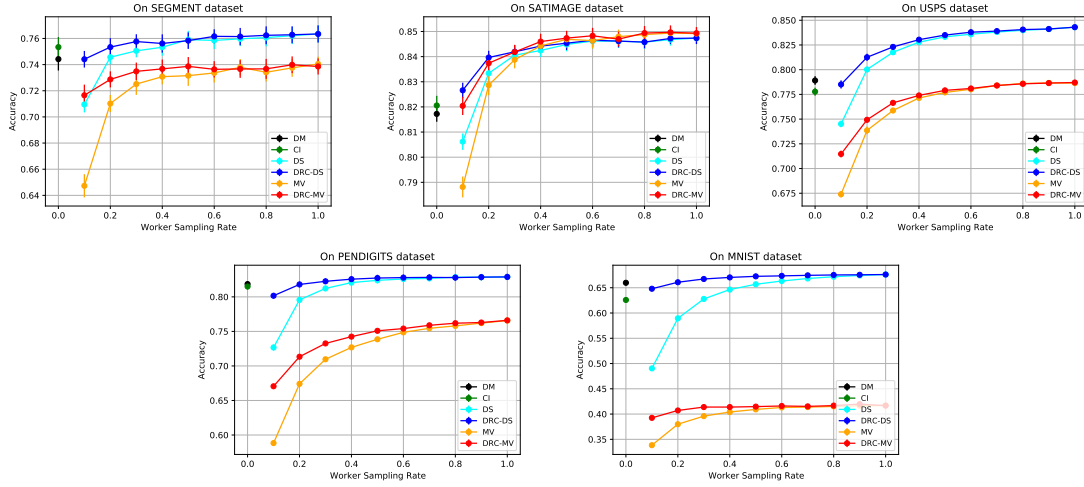
Figure 2: Performances of DRC-DS, DRC-MV, DS, MV, and DM.

our theoretical understanding. Moreover, our DRC approaches work well with very few workers and then perform better with sampling rate increasing.

- Because CI and DM involve no worker cost, they are two nodes in the figures corresponding to $\pi = 0$. Due to high bias incurred, DM performs unstably across all datasets. It performs well in some cases, but poorly on Satimage and Usps.

- On all datasets, except Satimage, DS performs better than MV. However, performance comparison between DS and MV is out of scope of this paper. We are focusing on improving existing approaches with DRC method.

6.1.3 EFFECTIVENESS OF AIS AND AWS

To show effectiveness of AIS and AWS, we compare four methods: DRC-DS, DRC-AIS, DRC-AWS, and DRC-AWS-AIS, while DM is used as the baseline. AWS and AIS rules are expected to save a lot of worker cost, so logarithmic worker cost is used, where cost is defined as number of workers that per item use. For DRC-AIS and DRC-AWS-AIS, the confidence margin parameter ρ is set to be 0.03, 0.06, and 0.09. For DRC-AWS and DRC-AWS-AIS, the multiplier parameter λ is set to be 1, 2, 3, 4, 5, 7, 10, 15, 25, and 50.

With dashed black lines being performances of DM, results are shown in Figure 3, which have five observations:

- Compared with DRC-DS, all AIS/AWS approaches are able to save labeling cost while maintaining almost the same accuracy, which validates AIS and AWS do play key roles in improving inference accuracy and saving worker cost at the same time.

- Among all four methods, DRC-AWS-AIS enjoys the lowest worker cost, which means AIS and AWS can work together.

- There is an accuracy-cost tradeoff for all approaches. All performances can be improved by introducing more worker cost, that is, greater ρ, λ.
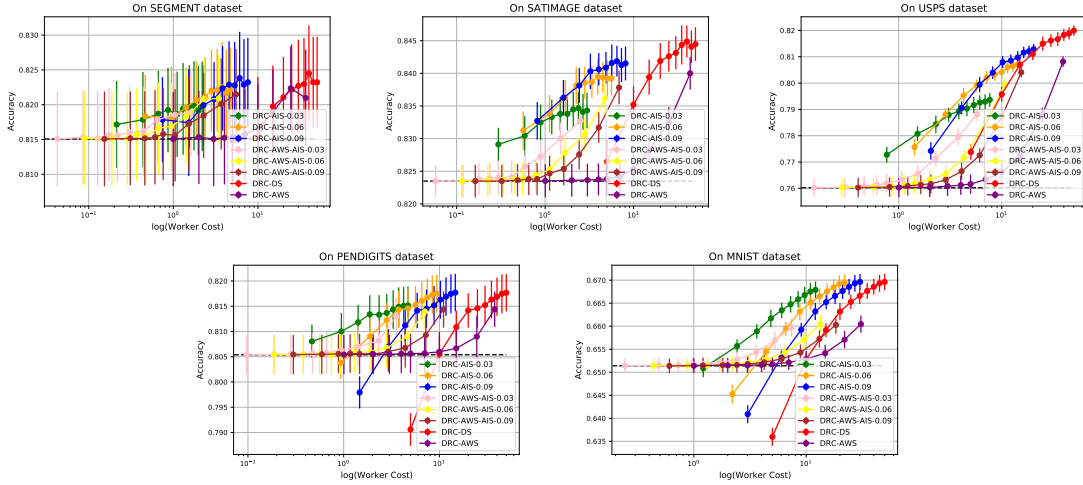
Figure 3: Performances of DRC-DS, DRC-AIS, DRC-AWS, and DRC-AWS-AIS.

- DRC-AWS and DRC-DS enjoy the same performance pattern while DRC-AWS-AIS and DRC-AIS have the same one. Specifically, because DM is applied at the first stage of algorithms, performances of DRC-AIS and DRC-AWS-AIS increase from the DM when worker cost is very small.

- No statistical conclusions can be made on Segment dataset due to high error bar.

6.1.4 Ablation Study on Model Misspecification

As mentioned in experimental settings, decision trees are used to generate the crowdsourcing datasets with item features, however, in real-world tasks, we have no idea of the label generating process. Therefore, model misspecification must be studied. In detail, Decision Trees (DT), Logistic Regression (LR), and Gaussian Naive Bayes (GNB) are used as supervised classifiers for DRC-DS, DRC-MV, and DM. $\pi = 0.5$ is set over source label matrix. All parameters of DT, LR, and GNB are set as default according to `sklearn` package. Other experimental settings are the same as before.

In Table 3, based on Wald's test, statistically better results are set in **bold** fonts. Decision trees perform only slightly better than other two classifiers, which is good because it show our approaches are able to work without knowing the label generating process. In real crowdsourcing problem, it is of a high chance that we have no idea of this process.

6.2 Real-world Experiments

Following synthetic experiments, we do algorithm comparison and study the effectiveness of AIS and AWS on real-world experiments.

6.2.1 Algorithm Comparison

We do experiments on three real-world datasets: Music Genre (Rodrigues et al., 2013), Dog (Zhou et al., 2012), and Rotten Tomatoes (Rodrigues et al., 2013). Because now we

Table 3: Performances of DRC-DS, DRC-MV, and DM with different supervised classifiers.

Dataset	Method	DT	LR	GNB
Segment	DRC-DS	0.7385 ± 0.0065	0.7425 ± 0.0063	0.7386 ± 0.0066
	DRC-MV	0.7731 ± 0.0053	0.7698 ± 0.0064	0.7677 ± 0.0053
	DM	0.7419 ± 0.0061	$\mathbf{0.7788} \pm 0.0084$	0.6969 ± 0.0084
Satimage	DRC-DS	0.8483 ± 0.0021	0.8452 ± 0.0028	0.8421 ± 0.0034
	DRC-MV	$\mathbf{0.8470} \pm 0.0025$	0.8416 ± 0.0029	0.8415 ± 0.0030
	DM	0.8478 ± 0.0016	0.8270 ± 0.0024	0.7940 ± 0.0033
Usps	DRC-DS	0.8323 ± 0.0024	0.8293 ± 0.0020	0.8260 ± 0.0017
	DRC-MV	0.8067 ± 0.0023	0.8022 ± 0.0029	0.7963 ± 0.0018
	DM	0.8315 ± 0.0021	$\mathbf{0.8374} \pm 0.0022$	0.7740 ± 0.0034
Pendigits	DRC-DS	$\mathbf{0.8196} \pm 0.0021$	0.8156 ± 0.0018	0.8156 ± 0.0020
	DRC-MV	$\mathbf{0.7359} \pm 0.0018$	0.7274 ± 0.0027	0.7266 ± 0.0023
	DM	0.8193 ± 0.0020	$\mathbf{0.8259} \pm 0.0024$	0.8223 ± 0.0017
Mnist	DRC-DS	0.6470 ± 0.0020	0.6426 ± 0.0022	0.5939 ± 0.0017
	DRC-MV	$\mathbf{0.5317} \pm 0.0013$	0.5240 ± 0.0015	0.4997 ± 0.0011
	DM	0.6471 ± 0.0019	0.6474 ± 0.0021	0.4240 ± 0.0035

Table 4: Statistics of real-world datasets.

Dataset	# item	# worker	# dimension	# class
Music	700	44	124	10
Dog	798	109	$5,376$	4
Tomato	$4,999$	203	$1,200$	2

have label matrix, we start from test dataset and item label matrix in Step 2 in Figure 1. Statistics of real-world datasets are shown in Table 4. Due to extreme few labels given by workers, two settings are different from synthetic experiments. First, only workers providing more than 40% labels are modeled and depth of decision trees is set to be 100 in order to ensure quality of surrogate labels. Second, worker sampling is conducted over existing labels given by worker, and the sampling rate goes from 0.1 to 0.5 with an interval of 0.1.

Table 5: Performances of compared algorithms on real world datasets. Based on Wald's test, statistically better results are shown in **bold** fonts.

Dataset	π	Algorithms and Performances			
		DM		**CI**	
	0.0	0.2286 ± 0.0134		$\mathbf{0.2734} \pm 0.0124$	
		MV	**DRC-MV**	**DS**	**DRC-DS**
	0.1	0.2557 ± 0.0087	$\mathbf{0.3449} \pm 0.0097$	0.2166 ± 0.0080	$\mathbf{0.2931} \pm 0.0113$
Music	0.2	0.3610 ± 0.0108	$\mathbf{0.4181} \pm 0.0110$	0.3006 ± 0.0102	$\mathbf{0.3300} \pm 0.0075$
	0.3	0.4493 ± 0.0105	$\mathbf{0.4824} \pm 0.0088$	0.3559 ± 0.0103	0.3716 ± 0.0107
	0.4	0.5097 ± 0.0091	0.5247 ± 0.0073	0.3926 ± 0.0108	0.3951 ± 0.0104
	0.5	0.5494 ± 0.0123	0.5601 ± 0.0085	0.4236 ± 0.0127	0.4159 ± 0.0109
		DM		**CI**	
	0.0	0.3637 ± 0.0086		$\mathbf{0.4115} \pm 0.0102$	
		MV	**DRC-MV**	**DS**	**DRC-DS**
	0.1	0.5481 ± 0.0092	$\mathbf{0.5860} \pm 0.0068$	0.5703 ± 0.0098	$\mathbf{0.5904} \pm 0.0072$
Dog	0.2	0.6644 ± 0.0121	$\mathbf{0.6792} \pm 0.0091$	0.6875 ± 0.0080	0.6919 ± 0.0066
	0.3	0.7238 ± 0.0080	0.7286 ± 0.0066	0.7450 ± 0.0078	0.7431 ± 0.0084
	0.4	0.7538 ± 0.0073	0.7593 ± 0.0074	0.7786 ± 0.0061	0.7741 ± 0.0077
	0.5	0.7749 ± 0.0064	0.7746 ± 0.0069	0.7906 ± 0.0069	0.7872 ± 0.0059
		DM		**CI**	
	0.0	0.5196 ± 0.0041		$\mathbf{0.5373} \pm 0.0036$	
		MV	**DRC-MV**	**DS**	**DRC-DS**
	0.1	0.6263 ± 0.0051	$\mathbf{0.6368} \pm 0.0037$	0.6311 ± 0.0033	$\mathbf{0.6425} \pm 0.0037$
Tomato	0.2	0.7134 ± 0.0034	0.7169 ± 0.0041	0.7285 ± 0.0035	0.7329 ± 0.0038
	0.3	0.7637 ± 0.0040	0.7663 ± 0.0039	0.7925 ± 0.0027	0.7924 ± 0.0034
	0.4	0.8025 ± 0.0030	0.8046 ± 0.0031	0.8374 ± 0.0021	0.8361 ± 0.0026
	0.5	0.8294 ± 0.0036	0.8295 ± 0.0035	0.8666 ± 0.0030	0.8629 ± 0.0030

For algorithm comparison, results are shown in Table 5, which have four observations:

- Given the same worker sampling rate, DRC-MV works better than MV and DRC-DS is better than DS, especially in low sampling rate cases. It shows our DRC approaches work in practice.

- As the sampling rate increasing, performances of DRC approaches converges to non-DRC approaches, which matches our understanding according to theorems.

- There are large improvements on Music and Dog datasets, while small improvements on Tomato datasets, potentially due to number of classes being too small. In other words, binary classification remains a easy task on Tomato dataset for workers. Bn contrast, it shows our DRC apporach is able to help in difficult multi-class classification problems.

- Compared between two zero cost methods, CI performs better than DM, which shows the effectiveness of supervised learners.
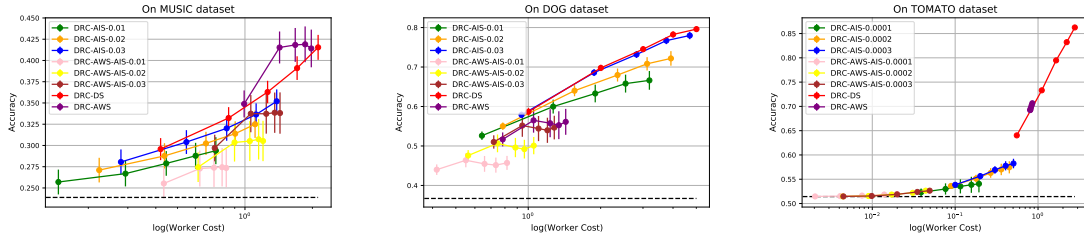
6.2.2 Effectiveness of AIS and AWS

Figure 4: Performances of DRC-AIS, DRC-AWS, DRC-AWS-AIS, and DRC-DS on real-world datasets.

Similar to synthetic experiments, we do experiments of DRC-AWS-AIS, DRC-AIS, DRC-AWS, and DRC-DS on real-world datasets. λ is set to be $1, 2, 4, 7, 10$ and ρ is set in Figure 4. Other settings remain the same as above. There are four observations from results shown in Figure 4:

- On Music dataset, DRC-AIS-0.01 enjoys the least worker cost, but with poor accuracy. DRC-AWS performs better than DRC-DS, both in cost and accuracy, which shows effectivenss of our weight-clipping technique.

- On Dog dataset, DRC-AWS-AIS enjoys the least worker cost, but performances increase quickly with DRC-AIS and DRC-DS.

- On Tomato dataset, it's hard to break the accuracy-cost tradeoff using AIS or ASW as the DRC-DS performance increases rapidly with more workers.

- There is an accuracy-cost tradeoff for all approaches on all datasets. For practical use, ρ is suggested to set from 0.0001 to 0.03, small for easy tasks, for example, binary classification, while large for multi-class classification. λ is suggested to set from 1 to c, where c is the average labels received per item. Small λ leads to low cost and low accuracy while large λ results in high cost and high accuracy.

7. Conclusion

We formulate crowdsourcing as a statistical estimation problem and propose a new approach DRC to address it where worker imitation and doubly robust estimation are used. DRC can work with any base models such as Dawid-Skene model and majority voting and improve their performance. With adaptive item/worker selection, our proposed approaches are able to achieve nearly the same accuracy of using all workers but with less worker cost. In the future, there are many problems worth trying. Since item features are helpful for crowdsourcing problems, worker features can be taken into consideration as well. Also, if there are new workers joining the project, it needs special considerations.

Acknowledgments

References

Heejung Bang and James M Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005.

Steve Branson, Grant Van Horn, and Pietro Perona. Lean crowdsourcing: Combining humans and machines in an online system. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7474–7483, 2017.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27, 2011.

Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied Statistics*, 28(1):20–28, 1979.

Miroslav Dudík, Dumitru Erhan, John Langford, and Lihong Li. Doubly robust policy evaluation and optimization. *Statistical Science*, 29(4):485–511, 2014.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, Shujie Liu, Tie-Yan Liu, Renqian Luo, Arul Menezes, Tao Qin, Frank Seide, Xu Tan, Fei Tian, Lijun Wu, Shuangzhi Wu, Yingce Xia, Dongdong Zhang, Zhirui Zhang, and Ming Zhou. Achieving human parity on automatic chinese to english news translation. arXiv preprint arXiv:1803.05567, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

Jonathan J Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994.

Hideaki Imamura, Issei Sato, and Masashi Sugiyama. Analysis of minimax error rate for crowdsourcing and its application to worker clustering model. In *Proceedings of the 35th International Conference on Machine Learning*, pages 2147–2156, 2018.

Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 652–661, 2016.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems 26*, pages 315–323, 2013.

Ashish Khetan, Zachary C Lipton, and Animashree Anandkumar. Learning from noisy singly-labeled data. In *Proceedings of the International Conference on Learning Representations*, 2018.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2012.

Jungseul Ok, Sewoong Oh, Yunhun Jang, Jinwoo Shin, and Yung Yi. Iterative bayesian learning for crowdsourced regression. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, pages 1486–1495, 2019.

Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. Learning from multiple annotators: distinguishing good from random labelers. *Pattern Recognition Letters*, 34(12): 1428–1436, 2013.

Andrea Rotnitzky and James M Robins. Semiparametric regression estimation in the presence of dependent censoring. *Biometrika*, 82(4):805–820, 1995.

Carl-Erik Särndal, Bengt Swensson, and Jan Wretman. *Model assisted survey sampling*. Springer Science & Business Media, 2003.

Nihar B Shah, Sivaraman Balakrishnan, and Martin J Wainwright. A permutation-based model for crowd labeling: Optimal estimation and robustness. *IEEE Transactions on Information Theory*, 67(6):4162–4184, 2020.

Victor S Sheng, Foster Provost, and Panagiotis G Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 614–622, 2008.

Grant Van Horn, Steve Branson, Scott Loarie, Serge Belongie, and Pietro Perona. Lean multiclass crowdsourcing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2714–2723, 2018.

Chong Wang, Xi Chen, Alexander J Smola, and Eric P Xing. Variance reduction for stochastic gradient optimization. In *Advances in Neural Information Processing Systems 26*, pages 181–189, 2013.

Yu-Xiang Wang, Alekh Agarwal, and Miroslav Dudik. Optimal and adaptive off-policy evaluation in contextual bandits. In *Proceedings of the 34th International Conference on Machine Learning*, pages 3589–3597, 2017.

Peter Welinder, Steve Branson, Pietro Perona, and Serge J Belongie. The multidimensional wisdom of crowds. In *Advances in Neural Information Processing Systems 23*, pages 2424–2432, 2010.

Wayne Xiong, Lingfeng Wu, Fil Alleva, Jasha Droppo, Xuedong Huang, and Andreas Stolcke. The microsoft 2017 conversational speech recognition system. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5934–5938, 2018.

Yuchen Zhang, Xi Chen, Dengyong Zhou, and Michael I Jordan. Spectral methods meet em: A provably optimal algorithm for crowdsourcing. *Journal of Machine Learning Research*, 17(1):3537–3580, 2016.

Dengyong Zhou, Sumit Basu, Yi Mao, and John C. Platt. Learning from the wisdom of crowds by minimax entropy. In *Advances in Neural Information Processing Systems 25*, pages 2195–2203, 2012.

Dengyong Zhou, Qiang Liu, John C Platt, Christopher Meek, and Nihar B Shah. Regularized minimax conditional entropy for crowdsourcing. arXiv preprint arXiv:1503.07240, 2015.