Crowdsourcing with Arbitrary Adversaries

Matthäus Kleindessner [1] **Pranjal Awasthi** [1]

Abstract

Most existing works on crowdsourcing assume that the workers follow the Dawid-Skene model, or the one-coin model as its special case, where every worker makes mistakes independently of other workers and with the same error probability for every task. We study a significant extension of this restricted model. We allow almost half of the workers to deviate from the one-coin model and for those workers, their probabilities of making an error to be task-dependent and to be arbitrarily correlated. In other words, we allow for arbitrary adversaries, for which not only error probabilities can be high, but which can also perfectly collude. In this adversarial scenario, we design an efficient algorithm to consistently estimate the workers' error probabilities.

1. Introduction

Crowdsourcing is an omnipresent phenomenon: it has emerged as an integral part of the machine learning pipeline in recent years, and one reason for the great advances in deep learning is the presence of large data sets that have been labeled by the crowd (e.g., Deng et al., 2009; Krizhevsky, 2009). Crowdsourcing is also at the heart of peer grading systems (e.g., Alfaro & Shavlovsky, 2014), which help with rising enrollment at universities, and online rating systems (e.g., Liao et al., 2014), which many of us rely on when choosing the next restaurant, to provide just a few examples.

A crowdsourcing scenario consists of a set of workers and a set of tasks that need to be solved. A data curator utilizing crowdsourcing can aim at estimating various quantities of interest. The first goal might be to estimate the true labels or answers for the tasks at hand. Typically, additional constraints are involved here such as a worker not being willing to solve too many tasks and the data curator wanting to get high-quality labels at a low price. The canonical example of this case is the Amazon Mechanical Turk™. There one cannot track specific workers as they are fleeting. However, in scenarios such as peer grading or online rating systems, a second goal might be to estimate worker qualities, especially if workers can be reused at a later time.

In a seminal paper, Dawid & Skene (1979) proposed a formal model that involves worker quality parameters for crowdsourcing scenarios in the context of classification. The Dawid-Skene model has become a standard theoretical framework and has led to a flurry of research over the past few years (Liu et al., 2012; Raykar & Yu, 2012; Li et al., 2013; Gao et al., 2016; Zhang et al., 2016; Khetan et al., 2017), in particular in its special symmetric form usually referred to as one-coin model (Ghosh et al., 2011; Karger et al., 2011a;b; Dalvi et al., 2013; Gao & Zhou, 2013; Karger et al., 2014; Bonald & Combes, 2017; Ma et al., 2017). In its general form for binary classification problems, the Dawid-Skene model assumes that for each worker, the probability of providing the wrong label only depends on the true label of the task, but not on the task itself. Moreover, given the true label, the responses provided by different workers are independent. The one-coin model additionally assumes that for each worker, the probability of providing the wrong label is the same for both classes. We will formally introduce the one-coin model in Section 2. A discussion of prior work work is provided in Section 5 and Appendix A.

The crucial limitation of the Dawid-Skene and one-coin model is the assumption that workers' error probabilities are task-independent. In particular, this excludes the possibility of colluding adversaries (other than those that provide the wrong label all of the time), which might make these models a poor approximation of the real world encountered in such applications as peer grading or online rating. In this paper, we study a significant extension of the one-coin model that allows for arbitrary, highly colluding adversaries. We provide an algorithm for estimating the workers' error probabilities and prove that it asymptotically recovers the true error probabilities. Using our estimates of the error probabilities in weighted majority votes, we also provide strategies to estimate ground-truth labels of the tasks. Experiments on both synthetic and real data show that our approach clearly outperforms existing methods in the presence of adversaries.

[1] Department of Computer Science, Rutgers University, Piscataway Township, New Jersey, USA. Correspondence to: Matthäus Kleindessner <matthaeus.kleindessner@rutgers.edu>, Pranjal Awasthi <pranjal.awasthi@rutgers.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

2. Setup and problem formulation

We first describe a general model for crowdsourcing with non-adaptive workers and binary classification tasks: there are n workers $w_1, \ldots, w_n$ and an i.i.d. sample of m task-label pairs $((x_i, y_i))_{i=1}^m \sim D^m$, where D is a joint probability distribution over tasks $x \in \mathcal{X}$ and corresponding labels $y \in \{-1, +1\}$. There is a variable $g_{ij} \in \{0, 1\}$, $i \in [m]$, $j \in [n]$, indicating whether worker w_j is presented with task x_i (for $k \in \mathbb{N}$, we use $[k]$ to denote the set $\{1, \ldots, k\}$). If w_j is presented with x_i, that is $g_{ij} = 1$, w_j provides an estimate $w_j(x_i) \in \{-1, +1\}$ of the ground-truth label y_i. Let $A \in \{-1, 0, +1\}^{m \times n}$ be a matrix that stores all the responses collected from the workers: $A_{ij} = w_j(x_i)$ if $g_{ij} = 1$ and $A_{ij} = 0$ if $g_{ij} = 0$.

We assume that each worker w_j follows some (probabilistic or deterministic) strategy such that $w_j(x_i)$ only depends on x_i. In particular, given x_i, any two different workers' responses $w_j(x_i)$ and $w_k(x_i)$ and the ground-truth label y_i are independent. Let $\varepsilon_{w_j}(x, y) \in [0, 1]$ be the conditional error probability that, given x and y, $w_j(x)$ does not equal y, that is

$$\varepsilon_{w_j}(x, y) := \Pr_{w_j | (x,y)}[w_j(x) \neq y \,|\, (x, y)]. \qquad (1)$$

Note that the unconditional probability of $w_j(x)$ being incorrect, before seeing x and y, is given by

$$\Pr_{(x,y)\sim D, w_j}[w_j(x) \neq y] = \mathbb{E}_{(x,y)\sim D}[\varepsilon_{w_j}(x, y)] =: \varepsilon_{w_j}.$$

Now one may study the following questions:

(i) Given only the matrix A, how can we estimate the ground-truth labels $y_1, \ldots, y_m$?

(ii) Given only the matrix A, how can we estimate the workers' unconditional error probabilities $\varepsilon_{w_1}, \ldots, \varepsilon_{w_n}$?

(iii) If we can choose g_{ij} (either in advance of collecting workers' responses or adaptively while doing so), how should we choose it such that we can achieve (i) or (ii) with a minimum number of collected responses?

In case of $\varepsilon_{w_j}(x, y)$ as defined in (1) being constant on $\mathcal{X} \times \{-1, +1\}$, that is $\varepsilon_{w_j}(x, y) \equiv \varepsilon_{w_j}$, for all $j \in [n]$, our model boils down to what is usually referred to as the *one-coin model* (e.g., Szepesvari, 2015), for which (i) to (iii) have been studied extensively (see Section 5 and Appendix A for references and a detailed discussion). With this paper we initiate the study of a significant extension of the one-coin model. We will allow almost half of the workers to deviate from the one-coin model and for such a worker w_j, the conditional error probability $\varepsilon_{w_j}(x, y)$ to be a completely arbitrary random variable. In other words, we will allow for arbitrary adversaries, for which not only error probabilities can be high, but for which error probabilities can be arbitrarily correlated. We mainly study (ii) in this scenario. We then make use of existing results for the one-coin model to answer (i) satisfactorily for our purposes. We do not deal with (iii), but instead assume that g_{ij} has been specified in advance.

3. General outline of our approach

In this section we want to present the general outline of our approach. A key insight is that the unconditional probability of workers w_j and w_k being agreeing is given by

$$\Pr_{(x,y)\sim D, w_j, w_k}[w_j(x) = w_k(x)] = 1 - \varepsilon_{w_j} - \varepsilon_{w_k} + 2\varepsilon_{w_j}\varepsilon_{w_k} + 2\,\mathrm{Cov}_{(x,y)\sim D}[\varepsilon_{w_j}(x, y), \varepsilon_{w_k}(x, y)]. \qquad (2)$$

$\mathrm{Cov}_{(x,y)\sim D}[\varepsilon_{w_j}(x, y), \varepsilon_{w_k}(x, y)]$ denotes the covariance between random variables $\varepsilon_{w_j}(x, y)$ and $\varepsilon_{w_k}(x, y)$, that is

$$\mathrm{Cov}_{(x,y)\sim D}[\varepsilon_{w_j}(x, y), \varepsilon_{w_k}(x, y)] = \mathbb{E}_{(x,y)\sim D}[(\varepsilon_{w_j}(x, y) - \varepsilon_{w_j}) \cdot (\varepsilon_{w_k}(x, y) - \varepsilon_{w_k})].$$

A proof of (2) can be found in Appendix B. The probability on the left-hand side of (2) can be easily estimated from A by the ratio of the number of tasks that w_j and w_k agreed on to the number of tasks they were both presented with:

$$\Pr[w_j(x) = w_k(x)] \approx \frac{\sum_{i=1}^m g_{ij} g_{ik} \mathbb{1}\{A_{ij} = A_{ik}\}}{\sum_{i=1}^m g_{ij} g_{ik}} =: p_{jk}. \qquad (3)$$

This suggests to solve the system of equations

$$1 - \varepsilon_j - \varepsilon_k + 2\varepsilon_j\varepsilon_k + 2c_{jk} = p_{jk}, \quad 1 \leq j < k \leq n, \qquad (4)$$

in the unknowns ε_l, $l \in [n]$, and c_{jk}, $1 \leq j < k \leq n$, in order to obtain estimates of the workers' unconditional error probabilities $\varepsilon_{w_1}, \ldots, \varepsilon_{w_n}$. However, there is a catch: in general, the system (4) is not identifiable and has several solutions. We will assume that at least $\frac{n}{2} + 2$ of the workers follow the one-coin model and have error probabilities smaller than one half. A worker w_j following the one-coin model implies

$$\mathrm{Cov}_{(x,y)\sim D}[\varepsilon_{w_j}(x, y), \varepsilon_{w_k}(x, y)] = 0, \quad \forall k \neq j, \qquad (5)$$

and hence under this assumption we can restrict the search for solutions of (4) to ε_l, $l \in [n]$, and c_{jk}, $1 \leq j < k \leq n$, with the property that[1]

$$\exists L \subseteq [n] \text{ with } |L| \geq n/2 + 2 \text{ such that } \forall j \in L : (\varepsilon_j < 1/2 \;\wedge\; [\forall k \neq j : c_{jk} = 0]). \qquad (6)$$

[1] Throughout the paper, we set $c_{jk} = c_{kj}$ if $j > k$. We also assume $p_{jk} = p_{kj}$.

Note that we never assume to know which workers follow the one-coin model, which corresponds to using the existential quantifier for the set L in (6) rather than considering a "fixed" L. We can show that the system (4) has at most one solution with property (6). We also provide evidence that our assumption of $\frac{n}{2}+2$ of the workers following the one-coin model and having error probabilities smaller than one half is a necessary condition for guaranteeing the identifiability of system (4). If the workers satisfy our assumption and p_{jk} on the right-hand side of (4) are actually true agreement probabilities, then $\varepsilon_l = \varepsilon_{w_l}$ and $c_{jk} = \mathrm{Cov}[\varepsilon_{w_j}(x,y), \varepsilon_{w_k}(x,y)]$ is the unique solution of (4) that satisfies (6). But if p_{jk} are not exactly true agreement probabilities, there might be no solution of (4) with property (6) at all. We prove that if estimates p_{jk} are not too bad, we can solve (4) together with (6) approximately, and our approximate solution is guaranteed to be close to true error probabilities $\varepsilon_{w_1}, \dots, \varepsilon_{w_n}$ and covariances $\mathrm{Cov}[\varepsilon_{w_j}(x,y), \varepsilon_{w_k}(x,y)]$, $j < k$. This answers (ii) from Section 2 and is the main contribution of our paper:

Main result. *Assume that at least $\frac{n}{2}+2$ of the workers follow the one-coin model and have error probabilities not greater than $\gamma_{\mathrm{TR}} < \frac{1}{2}$. If $|\Pr[w_j(x) = w_k(x)] - p_{jk}| \leq \beta$ for all $j \neq k$ and β sufficiently small, we can compute estimates $\hat{\varepsilon}_{w_1}, \dots, \hat{\varepsilon}_{w_n}$ of $\varepsilon_{w_1}, \dots, \varepsilon_{w_n}$ such that*

$$|\varepsilon_{w_i} - \hat{\varepsilon}_{w_i}| \leq C(\gamma_{\mathrm{TR}}) \cdot \beta^{1/4}.$$

We answer (i) from Section 2 and provide two ways to predict ground-truth labels $y_1, \dots, y_m$ by taking weighted majority votes over the responses provided by the workers. In these majority votes, the weights depend on our estimates of true error probabilities $\varepsilon_{w_1}, \dots, \varepsilon_{w_n}$.

4. Details and analysis

4.1. Estimating agreement probabilities

If g_{ij} has been specified in advance, we have the following guarantee on the quality of the estimates p_{jk} (see (3)):

Lemma 1. *Assume $\sum_{i=1}^m g_{ij} g_{ik} > 0$, $j \neq k$. Let $\delta > 0$ and*

$$\beta_{jk} = \min \left\{ 1, \left[\ln(2n^2/\delta) / \left(2 \sum_{i=1}^m g_{ij} g_{ik} \right) \right]^{1/2} \right\}.$$

Then we have with probability at least $1-\delta$ over the sample $((x_i, y_i))_{i=1}^m$ that

$$|\Pr[w_j(x) = w_k(x)] - p_{jk}| \leq \beta_{jk}, \quad 1 \leq j < k \leq n.$$

Proof. A straightforward application of Hoeffding's inequality and the union bound yields the result. □

4.2. Identifiability and approximate solution

If all workers follow the one-coin model, that is $\varepsilon_{w_j}(x,y) \equiv \varepsilon_{w_j}$ for all $j \in [n]$, we have $\mathrm{Cov}_{(x,y)\sim D}[\varepsilon_{w_j}(x,y), \varepsilon_{w_k}(x,y)] = 0, \quad 1 \leq j < k \leq n,$ and system (4) reduces to

$$1 - \varepsilon_j - \varepsilon_k + 2\varepsilon_j \varepsilon_k = p_{jk}, \quad 1 \leq j < k \leq n, \tag{7}$$

in the unknowns ε_l, $l \in [n]$. It is well known that, in general, even (7) is not identifiable. For example, if $p_{jk} = 1$ for all $1 \leq j < k \leq n$, there are the two solutions $\varepsilon_l = 0$, $l \in [n]$, and $\varepsilon_l = 1$, $l \in [n]$, corresponding to either all perfect or all completely erroneous workers. On the other hand, the system (7) is identifiable if we assume that on average workers are better than random guessing, that is $\frac{1}{n}\sum_{j=1}^n \varepsilon_{w_j} < \frac{1}{2}$, and there are at least three informative workers with $\varepsilon_{w_j} \neq \frac{1}{2}$ (Bonald & Combes, 2017).

Clearly, these two conditions do not guarantee identifiability of the general system (4). The next lemma shows that even if we additionally assume half of the workers to follow the one-coin model, the system (4) is not identifiable. Here we only state an informal version of the lemma. A detailed version and its proof can be found in Appendix B.

Lemma 2. *There exists an instance of the system (4), where n is even, that has two different solutions. In both solutions, it holds that $\varepsilon_l < \frac{1}{2}$, $l \in [n]$. Furthermore:*

(a) In the first solution, $c_{jk} = 0$ for all $j \in [\frac{n}{2}]$ and $k \neq j$, and ε_l is small if $l \in [\frac{n}{2}]$ and big if $l \in [n] \setminus [\frac{n}{2}]$.

(b) In the second solution, $c_{jk} = 0$ for all $j \in [n] \setminus [\frac{n}{2}]$ and $k \neq j$, and ε_l is small if $l \in [n] \setminus [\frac{n}{2}]$ and big if $l \in [\frac{n}{2}]$.

We want to mention that a solution of (4) does not necessarily correspond to actual workers, that is given ε_l, $l \in [n]$, and c_{jk}, $1 \leq j < k \leq n$, there might be no collection of workers $w_1, \dots, w_n$ such that $\varepsilon_{w_l} = \varepsilon_l$ and $\mathrm{Cov}[\varepsilon_{w_j}(x,y), \varepsilon_{w_k}(x,y)] = c_{jk}$. By the Bhatia-Davis inequality (Bhatia & Davis, 2010) it holds that $\mathrm{Var}[\varepsilon_{w_j}(x,y)] \leq \varepsilon_{w_j} - \varepsilon_{w_j}^2$. Hence, a necessary condition for a solution to correspond to actual workers is that $|c_{jk}| \leq (\varepsilon_j - \varepsilon_j^2)^{1/2}(\varepsilon_k - \varepsilon_k^2)^{1/2}$ (in addition to $\varepsilon_l \in [0,1]$). The two solutions in Lemma 2 correspond to actual workers.

From now on we assume that at least $\frac{n}{2}+2$ workers follow the one-coin model and have error probabilities smaller than one half:[2]

Assumption A. There exists $L \subseteq [n]$ with $|L| \geq n/2 + 2$ such that for all $j \in L$, the worker w_j follows the one-coin model with error probability $\varepsilon_{w_j} < 1/2$.

This corresponds to considering (4) together with the constraint (6). The system (4) together with (6) is identifiable:

Proposition 1. *There exists at most one solution of system (4) that has property (6).*

[2] All results of Section 4.2 hold true if we assume, more generally, the existence of $L \subseteq [n]$ with $|L| \geq \frac{n}{2} + 2$ such that (5) together with $\varepsilon_{w_j} < \frac{1}{2}$ holds for all $j \in L$.

Proof. Assuming there are two solutions $(\varepsilon_l^{S_1})_{l\in[n]}$, $(c_{jk}^{S_1})_{1\leq j<k\leq n}$ and $(\varepsilon_l^{S_2})_{l\in[n]}$, $(c_{jk}^{S_2})_{1\leq j<k\leq n}$ with L_1 and L_2 satisfying (6), there have to be pairwise different $i_1, i_2, i_3 \in L_1 \cap L_2$. It is easy to see that $(\varepsilon_{i_1}^{S_1}, \varepsilon_{i_2}^{S_1}, \varepsilon_{i_3}^{S_1})$ and $(\varepsilon_{i_1}^{S_2}, \varepsilon_{i_2}^{S_2}, \varepsilon_{i_3}^{S_2})$ and consequently also all the other components of the two solutions have to coincide. Details can be found in Appendix B. □

If p_{jk} at the right-hand side of (4) are true agreement probabilities, the true error probabilities $\varepsilon_{w_1}, \ldots, \varepsilon_{w_n}$ and covariances $\mathrm{Cov}[\varepsilon_{w_j}(x,y), \varepsilon_{w_k}(x,y)]$, $j<k$, make up the unique solution of (4) that satisfies (6), but if p_{jk} are not exactly true agreement probabilities, there might be no solution of (4) that satisfies (6) at all. Our goal is then to find a solution of (4) that satisfies (6) approximately and to show that our approximate solution has to be close to $\varepsilon_{w_1}, \ldots, \varepsilon_{w_n}$ and $\mathrm{Cov}[\varepsilon_{w_j}(x,y), \varepsilon_{w_k}(x,y)]$, $j<k$. As a first step towards this goal we need a generalization of Proposition 1:

Proposition 2. *Let $\gamma < 1/2$ and $\nu < 1/8 - \gamma/2 + \gamma^2/2$. If there exist two solutions $(\varepsilon_l^{S_i})_{l\in[n]}$, $(c_{jk}^{S_i})_{1\leq j<k\leq n}$, $i\in\{1,2\}$, of system (4) (where $p_{jk}\in[0,1]$) with the property that $\varepsilon_l^{S_i}\in[0,1]$, $l\in[n]$, and*

$$\begin{aligned}&\exists L_i \subseteq [n] \text{ with } |L_i| \geq n/2+2 \text{ such that}\\ &\qquad \forall j \in L_i : \left(\varepsilon_j^{S_i} \leq \gamma \ \wedge\ \left[\forall k \neq j : |c_{jk}^{S_i}| \leq \nu\right]\right),\end{aligned} \tag{8}$$

then

$$\left|\varepsilon_l^{S_1} - \varepsilon_l^{S_2}\right| \leq G(\gamma,\nu)\sqrt{\nu}, \quad \left|c_{jk}^{S_1} - c_{jk}^{S_2}\right| \leq 3G(\gamma,\nu)\sqrt{\nu}$$

for $l\in[n]$, $j<k$, where $G(\gamma,\nu)\to G(\gamma)>0$ as $\nu\to 0$.

The proof of Proposition 2, which provides an explicit expression for $G(\gamma,\nu)$, can be found in Appendix B.

In a next step, we assume that we are given pairwise different $i_1, i_2, i_3 \in [n]$ such that $w_{i_1}, w_{i_2}, w_{i_3}$ follow the one-coin model with $\varepsilon_{w_{i_1}}, \varepsilon_{w_{i_2}}, \varepsilon_{w_{i_3}} < 1/2$. In this case, assuming that estimates p_{jk} are close to true agreement probabilities, we can construct a solution of (4) that is guaranteed to be close to the true error probabilities and covariances (and hence approximately satisfies (6)). This is made precise in the next lemma (its proof can be found in Appendix B).

Lemma 3. *Let $\gamma_{\mathrm{TR}} < 1/2$ and consider the system (4) with $p_{jk}^{\mathrm{TR}}\in[0,1]$ as right-hand side. Assume there exists a solution*[3] *$(\varepsilon_l^{\mathrm{TR}})_{l\in[n]}$, $(c_{jk}^{\mathrm{TR}})_{1\leq j<k\leq n}$ with the property that $\varepsilon_l^{\mathrm{TR}}\in[0,1]$ and*

$$\begin{aligned}&\exists L^{\mathrm{TR}} \subseteq [n] \text{ with } |L^{\mathrm{TR}}| \geq n/2+2 \text{ such that}\\ &\qquad \forall j \in L^{\mathrm{TR}} : \left(\varepsilon_j^{\mathrm{TR}} \leq \gamma_{\mathrm{TR}} \ \wedge\ \left[\forall k \neq j : c_{jk}^{\mathrm{TR}} = 0\right]\right).\end{aligned} \tag{9}$$

Now consider the system (4) with $p_{jk}\in[0,1]$ as right-hand side. Assume that $|p_{jk}^{\mathrm{TR}} - p_{jk}| \leq \beta$ for all $j\neq k$, where β satisfies $\beta < 1/2 - 2\gamma_{\mathrm{TR}} + 2\gamma_{\mathrm{TR}}^2$. Let $i_1, i_2, i_3 \in [n]$ be pairwise different and set

$$\begin{aligned}&B := -2 + 4p_{i_1 i_3},\\ &C := 1 + 2p_{i_1 i_2}p_{i_2 i_3} - p_{i_1 i_2} - p_{i_1 i_3} - p_{i_2 i_3},\\ &\varepsilon_{i_2}^R := \frac{1}{2} - \frac{\sqrt{B+4C}}{2\sqrt{B}}, \qquad \varepsilon_{i_2}^S := \min(\gamma_{\mathrm{TR}}, \max(0, \varepsilon_{i_2}^R))\end{aligned} \tag{10}$$

and for all $l \neq i_2$ and for all $1\leq j<k\leq n$

$$\begin{aligned}&\varepsilon_l^R := \frac{p_{i_2 l} - 1 + \varepsilon_{i_2}^S}{2\varepsilon_{i_2}^S - 1},\\ &\varepsilon_l^S := \begin{cases}\min(\gamma_{\mathrm{TR}}, \max(0, \varepsilon_l^R)) & \text{if } l \in \{i_1, i_3\}\\ \min(1, \max(0, \varepsilon_l^R)) & \text{if } l \notin \{i_1, i_3\}\end{cases},\\ &c_{jk}^S := \frac{p_{jk} - (1 - \varepsilon_j^S - \varepsilon_k^S + 2\varepsilon_j^S\varepsilon_k^S)}{2}.\end{aligned} \tag{11}$$

If all expressions are defined (i.e., $B>0$, $B+4C\geq 0$ and $\varepsilon_{i_2}^S \neq \frac{1}{2}$), then $(\varepsilon_l^S)_{l\in[n]}$, $(c_{jk}^S)_{1\leq j<k\leq n}$ is a solution of (4) with p_{jk} as right-hand side. If $i_1, i_2, i_3 \in L^{\mathrm{TR}}$, then all expressions are defined and

$$\begin{aligned}&\left|\varepsilon_l^{\mathrm{TR}} - \varepsilon_l^S\right| \leq H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta}, \quad l\in[n],\\ &\left|c_{jk}^{\mathrm{TR}} - c_{jk}^S\right| \leq 3H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta} + \beta/2, \quad j<k,\end{aligned} \tag{12}$$

where $H(\gamma_{\mathrm{TR}},\beta)\to H(\gamma_{\mathrm{TR}})>0$ as $\beta\to 0$.

In Lemma 3, for constructing the solution $(\varepsilon_l^S)_{l\in[n]}$, $(c_{jk}^S)_{1\leq j<k\leq n}$ as defined in (10) and (11) we need to know $\gamma_{\mathrm{TR}} < 1/2$, which is an upper bound on the error probabilities of at least $\frac{n}{2}+2$ workers that follow the one-coin model. In practice, we might choose γ_{TR} depending on the difficulty of the tasks or simply set it conservatively, for example as $\gamma_{\mathrm{TR}} = 0.45$. If $i_1, i_2, i_3 \in L^{\mathrm{TR}}$, then (12) implies that $(\varepsilon_l^S)_{l\in[n]}$, $(c_{jk}^S)_{1\leq j<k\leq n}$ satisfies (8) with

$$\gamma = \gamma_{\mathrm{TR}} + H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta}, \quad \nu = 3H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta} + \beta/2. \tag{13}$$

If we know the value of β (using Lemma 1, we easily obtain an upper bound β that holds with high probability), we can compute these quantities. This suggests the following strategy for obtaining estimates of $\varepsilon_{w_1}, \ldots, \varepsilon_{w_n}$ and $\mathrm{Cov}[\varepsilon_{w_j}(x,y), \varepsilon_{w_k}(x,y)]$, $j<k$: we sample pairwise different $i_1, i_2, i_3 \in [n]$ uniformly at random and construct $(\varepsilon_l^S)_{l\in[n]}$, $(c_{jk}^S)_{1\leq j<k\leq n}$ as defined in (10) and (11). If one of the expressions is not defined, we can immediately discard (i_1, i_2, i_3). Otherwise, we check whether $(\varepsilon_l^S)_{l\in[n]}$, $(c_{jk}^S)_{1\leq j<k\leq n}$ satisfies (8) with γ and ν as specified in (13). If it does, since $(\varepsilon_l^{\mathrm{TR}})_{l\in[n]}$, $(c_{jk}^{\mathrm{TR}} + (p_{jk} - p_{jk}^{\mathrm{TR}})/2)_{1\leq j<k\leq n}$ is a solution of (4) with p_{jk} as right-hand side that satisfies

[3] By Proposition 1, this solution is unique.

property (8) too, Proposition 2 guarantees that

$$\left|\varepsilon_l^{\mathrm{TR}} - \varepsilon_l^S\right| \le \sqrt{3H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta} + \frac{\beta}{2}} \cdot G\left(\gamma_{\mathrm{TR}} + H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta}, 3H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta} + \frac{\beta}{2}\right) \sim \beta^{1/4} \tag{14}$$

for all $l \in [n]$ and a similar bound on $|c_{jk}^{\mathrm{TR}} - c_{jk}^S|$, $j < k$. If $(\varepsilon_l^S)_{l\in[n]}$, $(c_{jk}^S)_{1\le j<k\le n}$ does not satisfy (8), we discard (i_1, i_2, i_3) and start anew. Note that under our Assumption A, the probability of choosing i_1, i_2, i_3 such that $i_1, i_2, i_3 \in L^{\mathrm{TR}}$ is greater than $1/8$. In expectation we have to discard (i_1, i_2, i_3) for not more than eight times before finding a solution that satisfies (8) and hence (14).

Assuming that every worker is presented with every task, that is $g_{ij} = 1$ for all $i \in [m]$ and $j \in [n]$, it follows from Lemma 1 and (14) that m has to scale as $\ln(n^2/\delta)/\rho^8$ in order that the described strategy is guaranteed to yield, with probability at least $1 - \delta$, estimates $\varepsilon_1^S, \ldots, \varepsilon_n^S$ satisfying $\left|\varepsilon_l^{\mathrm{TR}} - \varepsilon_l^S\right| \le \rho$, $l \in [n]$. This is significantly larger than the rate $m \sim \ln(n^2/\delta)/\rho^2$ required by the TE algorithm, which solves the estimation problem for the error probabilities in the one-coin model and is claimed to be minimax optimal (Bonald & Combes, 2017). We suspect that our rate with its dependence on ρ^{-8} is not optimal and consider it to be an interesting follow-up question to study the minimax rate for our extension of the one-coin model.

Although the convergence rate that we can guarantee for the described strategy is slow, we might still hope that the strategy performs better in practice. However, there is an issue that we have to overcome. Unless β is very small, γ and ν as specified in (13) are too big for being meaningful, that is any solution $(\varepsilon_l^S)_{l\in[n]}$, $(c_{jk}^S)_{1\le j<k\le n}$ as defined in (10) and (11) will satisfy (8) with these values. We will not discard any (i_1, i_2, i_3), regardless of whether $i_1, i_2, i_3 \in L^{\mathrm{TR}}$ holds or not. We deal with this issue by adapting the strategy as follows: let $P \subseteq \{(i_1, i_2, i_3) : i_1, i_2, i_3 \in [n] \text{ pairwise different}\}$. For every $p = (i_1, i_2, i_3) \in P$, we construct $(\varepsilon_l^S(p))_{l\in[n]}$, $(c_{jk}^S(p))_{1\le j<k\le n}$ as defined in (10) and (11). We set $Q^p = [n]$ unless γ as specified in (13) is smaller than one, in which case we set $Q^p = \{l \in [n] : \varepsilon_l^S(p) \le \gamma\}$ and discard any solution $(\varepsilon_l^S(p))_{l\in[n]}$, $(c_{jk}^S(p))_{1\le j<k\le n}$ for which $|Q^p| < \frac{n}{2} + 2$. Let ν^p be the $\lceil \frac{n}{2} + 2 \rceil$-th smallest element of $\{\max_{k\in[n]\setminus\{l\}} |c_{lk}^S(p)| : l \in Q^p\}$. Then we finally return the solution $(\varepsilon_l^S(p_0))_{l\in[n]}$, $(c_{jk}^S(p_0)))_{1\le j<k\le n}$ for which ν^p is smallest, that is $p_0 = \operatorname{argmin}_p \nu^p$.

If γ is small enough, it follows from Proposition 2 that

$$\left|\varepsilon_l^{\mathrm{TR}} - \varepsilon_l^S(p_0)\right| \le \sqrt{\max\{\nu^{p_0}, \beta/2\}} \cdot G\left(\gamma_{\mathrm{TR}} + H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta}, \max\{\nu^{p_0}, \beta/2\}\right). \tag{15}$$

Note that if P contains at least one triple of indices $i_1, i_2, i_3 \in L^{\mathrm{TR}}$, then $\nu^{p_0} \le 3H(\gamma_{\mathrm{TR}},\beta)\sqrt{\beta} + \frac{\beta}{2}$, so that the guarantee (15) is at least as good as (14). We also expect ν^{p_0} to be smaller the larger P is. Hence, we should choose P as large as we can afford due to computational reasons, but in practice, there is one more aspect that we have to consider. Depending on how g_{ij} has been chosen, there might be workers w_j and w_k that were presented with only a few common tasks or no common tasks at all. In this case, the estimate p_{jk} of the agreement probability between w_j and w_k is only poor and there is no uniform bound β on $|p_{jk}^{\mathrm{TR}} - p_{jk}|$ (where p_{jk}^{TR} are true agreement probabilities). We can deal with this aspect by choosing P in a way such that for all $p \in P$, all estimates p_{jk} that are involved in the computation of $(\varepsilon_l^S(p))_{l\in[n]}$ are somewhat reliable. We present a concrete implementation of this in Algorithm 1 below.

4.3. Predicting ground-truth labels

Once we have estimates $\hat{\varepsilon}_{w_1}, \ldots, \hat{\varepsilon}_{w_n}$ of the true error probabilities $\varepsilon_{w_1}, \ldots, \varepsilon_{w_n}$, we predict ground-truth labels y_i by taking a weighted majority vote over the responses collected for the task x_i. Our estimate for y_i is given by

$$\hat{y}_i = \operatorname{sign}\left\{\sum\nolimits_{l=1}^{n} f(\hat{\varepsilon}_{w_l}) \cdot A_{il}\right\}, \tag{16}$$

where $f : [0, 1] \to [-\infty, +\infty]$. Ties are broken uniformly at random. We consider two choices for the function f.

It is well-known that if all workers follow the one-coin model with known error probabilities $\varepsilon_{w_1}, \ldots, \varepsilon_{w_n}$, ground-truth labels are balanced, that is $\Pr_{(x,y)\sim D}[y = +1] = \Pr_{(x,y)\sim D}[y = -1]$, and g_{ij} are independent Bernoulli random variables with common success probability $\alpha > 0$, then the optimal estimator for the ground-truth label y_i is given by the weighted majority vote (16) with $f(\hat{\varepsilon}_{w_l})$ replaced by $f(\varepsilon_{w_l}) = \ln((1 - \varepsilon_{w_l})/\varepsilon_{w_l})$ (Nitzan & Paroush, 1982; Berend & Kontorovich, 2015; Bonald & Combes, 2017). Hence, a common approach for the one-coin model is to first estimate the true error probabilities and then to estimate ground-truth labels by using the majority vote (16) with $f(\hat{\varepsilon}_{w_l}) = \ln((1 - \hat{\varepsilon}_{w_l})/\hat{\varepsilon}_{w_l})$ (Bonald & Combes, 2017; Ma et al., 2017). We propose to use the same majority vote, but restricted to answers from workers that we believe to follow the one-coin model. Using the notation from Section 4.2, this means that we set $f(\hat{\varepsilon}_{w_l}) = \ln((1 - \hat{\varepsilon}_{w_l})/\hat{\varepsilon}_{w_l})$ for $l \in Q^{p_0}$ with $\max_{k\in[n]\setminus\{l\}} |c_{lk}^S(p_0)| \le \nu^{p_0}$ and $f(\hat{\varepsilon}_{w_l}) = 0$ otherwise.

Alternatively, we suggest to set $f(\hat{\varepsilon}_{w_l}) = 1 - 2\hat{\varepsilon}_{w_l}$ for $l \in [n]$. With this choice of f we make use of the responses provided by all workers. The same choice has been used for the one-coin model too (Dalvi et al., 2013). A third option would be to set $f(\hat{\varepsilon}_{w_l}) = 1 - 2\hat{\varepsilon}_{w_l}$ for $l \in Q^{p_0}$ with $\max_{k\in[n]\setminus\{l\}} |c_{lk}^S(p_0)| \le \nu^{p_0}$ and $f(\hat{\varepsilon}_{w_l}) = 0$ otherwise, but we do not consider this choice any further.

4.4. Algorithm

In the interests of clarity, we present our approach as self contained Algorithm 1. Choosing P as the set of triples such that involved pairs of workers have been provided with at least ten or three common tasks might seem somewhat arbitrary here. Indeed, one could introduce two parameters to the algorithm instead. Without optimizing for these parameters, we chose them as ten and three in all our experiments on real data, and hence we state Algorithm 1 as is.

Our analysis best applies to the setting of a full matrix A (or variables g_{ij} that are independent Bernoulli random variables with common success probability, as it is assumed by Bonald & Combes, 2017, for example). In this case, which we consider in our experiments on synthetic data, choosing P as stated in Algorithm 1 reduces to choosing P as the set of all triples of pairwise different indices. If the number of workers n is small, this is the best one can do. If n is large, it is infeasible to choose P as the set of all triples though since the running time of Algorithm 1 is in $\mathcal{O}(n^2(m+|P|))$. If n is large and A full, one should sample P uniformly at random. For $|P| \geq \ln\delta/\ln(7/8)$ our error guarantee (14) holds with probability at least $1-\delta$ then (compare with Section 4.2).

5. Related work

We briefly survey related work here. A complete discussion can be found in Appendix A. As discussed in Sections 1 and 2, in crowdsourcing one might be interested in estimating ground-truth labels and/or worker qualities given the response matrix A, but also in optimal task assignment. In their seminal paper, Dawid & Skene (1979) proposed an EM based algorithm to address the first two goals. Since then numerous works have followed addressing all three goals for the Dawid-Skene and one-coin model (Ghosh et al., 2011; Karger et al., 2011a;b; 2013; 2014; Dalvi et al., 2013; Gao & Zhou, 2013; Gao et al., 2016; Zhang et al., 2016; Bonald & Combes, 2017; Ma et al., 2017). There have also been efforts to study generalizations of the Dawid-Skene model (Jaffe et al., 2016; Khetan & Oh, 2016; Shah et al., 2016) as well as to explicitly deal with adversaries (Raykar & Yu, 2012; Jagabathula et al., 2017). However, none of the prior work can handle a number of arbitrary adversaries almost as large as the number of reliable workers as we do.

6. Experiments

On both synthetic and real data, we compared our proposed Algorithm 1 to straightforward majority voting for predicting labels (referred to as Maj) and the following methods from the literature: the spectral algorithms by Ghosh et al. (2011) (GKM), Dalvi et al. (2013) (RoE and EoR) and Karger et al. (2013) (KOS), the two-stage procedure by Zhang et al. (2016) (S-EM1 and S-EM10, where we run one or ten iterations of the EM algorithm) and the recent method by Bonald & Combes (2017) (TE). We used the Matlab implementation of KOS, S-EM1 and S-EM10 made available by Zhang et al. (2016). In our implementations of the other methods, we adapted GKM, RoE and EoR as to assume that the average error of the workers is smaller than one half rather than assuming that the error of the first worker is. We always called Algorithm 1 with parameters $\gamma_{\text{TR}} = 0.4$ and $\delta = 0.1$, which resulted in γ being set to 1

Algorithm 1

```
Input: crowdsourced labels stored in $A \in \{-1,0,+1\}^{m\times n}$, upper bound $0 < \gamma_{\text{TR}} < \frac{1}{2}$
  on the error probabilities of $\lceil \frac{n}{2}+2 \rceil$ workers that follow
  the one-coin model, confidence parameter $0 < \delta < 1$
Output: estimates $(\varepsilon^F_l)_{l\in[n]}$, $(c^F_{jk})_{j<k}$, $(\hat{y}_i)_{i\in[m]}$ of error probabilities, covariances and ground-truth labels

▶ Estimating agreement probabilities
set $g_{ij} = \mathbb{1}\{A_{ij} \neq 0\}$, $i \in [m]$, $j \in [n]$
set $q_{jk} = \sum_{i=1}^m g_{ij} g_{ik}$, $j,k \in [n]$
set $p_{jk}$ as in (3), $j,k \in [n]$ ($p_{jk} = \text{NaN}$ if $q_{jk} = 0$)

▶ Estimating error probabilities and covariances
set $\beta = \left[\ln(2n^2/\delta)/\left(2\min_{j,k\in[n]} q_{jk}\right)\right]^{1/2} \in (0, +\text{Inf}]$
set $\gamma$ as in (13)
if $\gamma \notin [0,1]$ then
  set $\gamma = 1$
end if
set $P = \{(i_1,i_2,i_3) : i_1,i_2,i_3 \in [n]$ pairwise different
  and $q_{jk} \geq 10$, $j,k \in \{i_1,i_2,i_3\}$, and $q_{i_2 j} \geq 3$, $j \neq i_2\}$
set $\nu_{\text{old}} = \text{Inf}$, $(\varepsilon^F_l)_{l\in[n]} = 0$, $(c^F_{jk})_{1\leq j<k\leq n} = 0$, $L = \emptyset$
for $(i_1,i_2,i_3) \in P$ do
  if not all expressions in (10) or (11) are defined then
    break
  end if
  compute $(\varepsilon^S_l)_{l\in[n]}$, $(c^S_{jk})_{1\leq j<k\leq n}$ as in (10) and (11)
  set $Q = \{l \in [n] : \varepsilon^S_l \leq \gamma\}$
  set $\nu = \lceil \frac{n}{2}+2 \rceil$-th smallest element of
      $\{\max_{k\in[n]\setminus\{l\}} |c^S_{lk}| : l \in Q\}$ ($\nu = \text{NaN}$ if $Q = \emptyset$)
  if $|Q| \geq \frac{n}{2}+2$ AND $\nu < \nu_{\text{old}}$ then
    set $(\varepsilon^F_l)_{l\in[n]} = (\varepsilon^S_l)_{l\in[n]}$, $(c^F_{jk})_{j<k} = (c^S_{jk})_{j<k}$
    set $L = \{l \in Q : \max_{k\in[n]\setminus\{l\}} |c^S_{lk}| \leq \nu\}$
    set $\nu_{\text{old}} = \nu$
  end if
end for

▶ Estimating ground-truth labels
set $f(\hat{\varepsilon}_{w_l}) = \ln((1-\hat{\varepsilon}_{w_l})/\hat{\varepsilon}_{w_l}) \in [-\text{Inf}, +\text{Inf}]$, $l \in L$,
  and $f(\hat{\varepsilon}_{w_l}) = 0$, $l \in [n] \setminus L$
(alternatively set $f(\hat{\varepsilon}_{w_l}) = 1 - 2\hat{\varepsilon}_{w_l}$, $l \in [n]$)
set $\hat{y}_i$ as in (16), $i \in [m]$
```

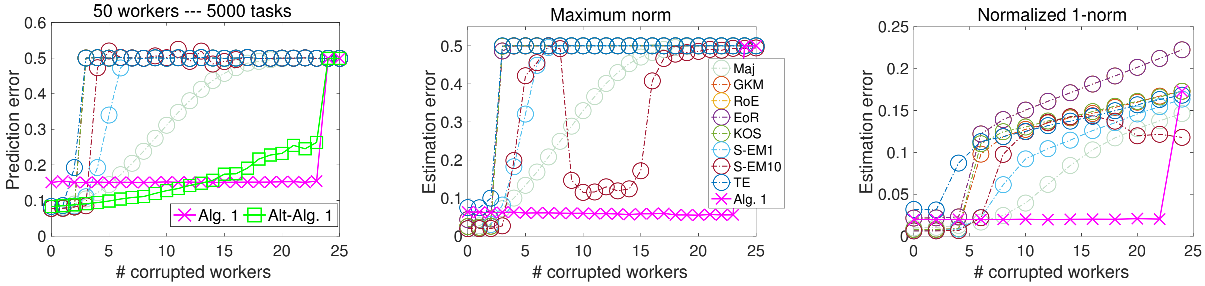

Figure 1. Synthetic data: prediction error and estimation error as a function of the number of corrupted workers.

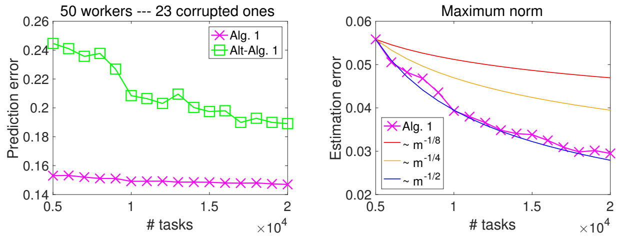

Figure 2. Synthetic data: prediction and estimation error of Algorithm 1 as a function of the number of tasks m.

in the execution of the algorithm in all our experiments. We refer to Algorithm 1 with the logarithmic weights in (16) as Alg. 1 and and with the linear weights as Alt-Alg. 1. In the following, all results are average results obtained from running an experiment for 100 times.

6.1. Synthetic data

In our first experiment, we consider $n = 50$ workers and $m = 5000$ tasks with balanced ground-truth labels. Every worker is presented with every task. For $0 \leq t \leq 25$, we choose t workers at random. These workers are corrupted workers that all provide the same random response to every task, which is incorrect with error probability 0.5. The remaining $n - t$ workers provide responses according to the one-coin model, where the error probability of each of these workers is 0.4. Figure 1 shows the prediction error for estimating ground-truth labels and the estimation error for estimating error probabilities in both the maximum norm and the normalized 1-norm for the various methods as a function of t. The prediction error is given by $\frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\{y_i \neq \hat{y}_i\}$ for ground-truth labels y_i and estimates $\hat{y}_i$ and the estimation error is given by $\max_{l\in[n]} |\varepsilon_{w_l} - \hat{\varepsilon}_{w_l}|$ or $\frac{1}{n}\sum_{l=1}^{n} |\varepsilon_{w_l} - \hat{\varepsilon}_{w_l}|$ for true error probabilities ε_{w_l} and estimates $\hat{\varepsilon}_{w_l}$. The methods Maj and KOS, by default, do not provide estimates of the workers' error probabilities. We adapt these two methods in order to return estimates of the error probabilities too as follows: if the method returns label estimates $\hat{y}_1, \ldots, \hat{y}_m$ and worker w_l provides responses $A_{1l}, \ldots, A_{ml} \neq 0$, then the method returns $\frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\{\hat{y}_i \neq A_{il}\}$ as estimate $\hat{\varepsilon}_{w_l}$ of ε_{w_l}.

Our Algorithm 1 is the only method that can handle up to $23 = \frac{n}{2} - 2$ corrupted workers (in accordance with our theoretical results). Its estimation error is constant as the number of corrupted workers increases from 0 to 23. Its prediction error depends on which weights we use in (16): the prediction error of Alg. 1 is constant in this range too, the one of Alt-Alg. 1 is slightly increasing. If only a few workers are corrupted, Alt-Alg.1 performs better than Alg. 1, while it is the other way round if more than 13 workers are corrupted. The methods from the literature predict ground-truth labels as badly as random guessing already in the presence of only six corrupted workers. All these methods are outperformed by majority voting. We do not have an explanation for the non-monotonic behavior of the estimation error of S-EM10 in the maximum norm. In Appendix C we present similar experiments, in which the error probability of the workers following the one-coin model is smaller or the error probabilities of the corrupted workers are less correlated. Still, the overall picture there is the same.

One might wonder whether one can combine the considered methods from the literature with one of the algorithms by Jagabathula et al. (2017) in order to first sort the corrupted workers out and then apply the method only to the remaining workers and their responses. However, those algorithms cannot deal with the corrupted workers considered in this experiment, which are perfectly colluding, at all. Even though provided with the correct number t of corrupted workers as input, when $t \geq 3$, the soft-penalty algorithm by Jagabathula et al. (2017) was not able to identify any of the corrupted workers in any of the 100 runs of the experiment.

In our next experiment, we study the convergence rate of Algorithm 1. We consider $n = 50$ workers, out of which $t = 23$ are corrupted in the same way as above. Figure 2 shows the prediction and estimation error of Algorithm 1 as a function of the number of tasks m varying from 5000 to 20000. The prediction error of Alg. 1 decreases only slightly as m increases, the prediction error of Alt-Alg. 1 decreases more significantly. Most interesting is the decay of the estimation error. Apparently, in this experiment it

Data set	Maj	GKM	RoE	EoR	KOS	S-EM1	S-EM10	TE	Alg. 1	Alt-Alg. 1
Bird	0.2407	0.2778	0.2778	0.2778	0.1111	0.1111	0.1019	0.1759	0.2963	0.2778
Dog	0.1883	0.2020	0.1834	0.1871	0.2069	0.1834	0.1772	0.9913	0.1921	0.1859
Duchenne	0.3802	0.3000	0.3125	0.3250	0.3813	0.3250	0.3562	0.3562	0.3062	0.2937
RTE	0.2562	0.4925	0.4937	0.1175	0.4000	0.1613	0.1025	0.2100	0.3638	0.2900
Temp	0.0976	0.5649	0.5693	0.0563	0.0671	0.0671	0.0628	0.0714	0.1991	0.0584
Web	0.1217	0.0249	0.0426	0.1014	0.0377	0.0931	0.0513	0.9955	0.0309	0.0611

Table 1. Real world data sets: prediction error of the various methods. The smallest value of each row is shown in red.

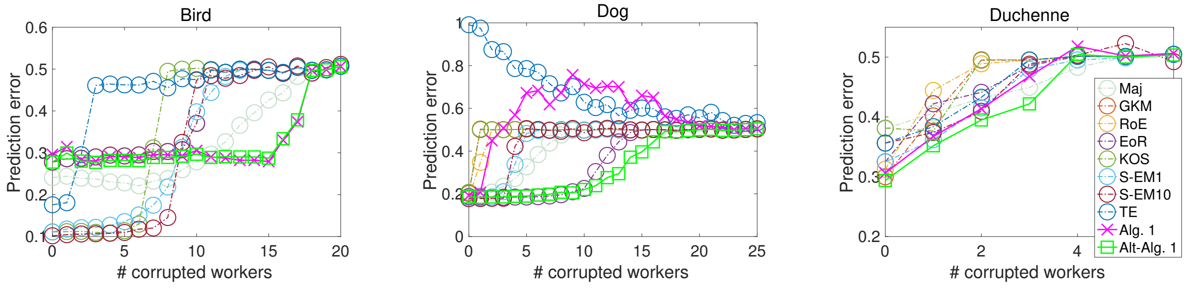

Figure 3. Real world data sets: prediction error of the various methods as a function of the number of corrupted workers.

decreases at a rate of $m^{-1/2}$ rather than at a rate of $m^{-1/8}$ as suggested by our upper bound (compare with Section 4.2).

6.2. Real data

We performed experiments on six publicly available data sets that are are commonly used in the literature (cf. Snow et al., 2008, Zhang et al., 2016, and Bonald & Combes, 2017). All six data sets come with ground truth labels for each task. For most of the data sets the matrix A, which stores the collected responses, is highly sparse. In order to reduce sparseness, we removed workers that provided fewer than 50 labels. For two of the data sets, we merged classes in order to end up with binary classification problems in the same way as Bonald & Combes (2017) did (Dog: $\{0, 2\}$ vs $\{1, 3\}$; Web: $\{0, 1, 2\}$ vs $\{3, 4\}$). Table 2 in Appendix C provides the characteristic values of the data sets. Note that only for the Bird data set every worker provided a label for every task whereas for the other ones A is still rather sparse. Figure 5 in Appendix C shows for each data set a histogram of the error probabilities of the workers (computed over those tasks that a worker was presented with). Figure 6 shows a heat map of the matrix $(|\text{Cov}[\varepsilon_{w_j}(x, y), \varepsilon_{w_k}(x, y)]|)_{j,k=1}^{n}$ (computed over those tasks that two workers were jointly presented with).

Table 1 shows the prediction error for the various methods and data sets. There is no method that performs best on all data sets. Overall, S-EM10 seems to be the method of choice. Our Algorithm 1 can compete with the other methods, and on four out of the six data sets, the prediction error of Alt-Alg. 1 is smaller or larger only by 0.01 than the prediction error of S-EM10. Alg. 1 performs slightly worse than Alt-Alg. 1. The poor performance of our method on the Bird data set might be explained by the fact that there the workers clearly deviate from our model: as Figure 6 shows, there are no $\frac{n}{2} + 2$ workers that follow the one-coin model.

We performed another experiments on these data sets by corrupting some of the workers (chosen at random). Like in the experiments of Section 6.1, the corrupted workers provide the same random response to every task. Figure 3 shows the prediction errors for the various methods and the first three data sets as functions of the number of corrupted workers. Similar plots for the other data sets are shown in Figure 7 in Appendix C. On none of the data sets, any method can handle more corrupted workers than Alt-Alg. 1.

7. Discussion

In this work, we studied an extension of the well-known one-coin model for crowdsourcing that allows for colluding adversaries. Our results show that even if almost half of the workers are adversarial, one can consistently estimate the workers' error probabilities with an efficient algorithm.

For future work, it would be interesting to relax the assumption that the reliable workers follow the one-coin model and to allow for task-dependent error probabilities also for them. It would also be interesting to see whether our approach can be extended to multiclass classification problems. Another direction concerns improving the sufficient rate $m \sim \rho^{-8}$, which we obtained for our algorithm for recovering worker qualities up to error ρ. In the absence of adversaries one can achieve a rate $m \sim \rho^{-2}$, and we would like to understand whether this gap is inherent or an artifact of our algorithm/proof. Finally, we wonder about the role of adaptive task assignment in our extension of the one-coin model.

Acknowledgements

This research is supported by a Rutgers Research Council Grant and a Center for Discrete Mathematics and Theoretical Computer Science (DIMACS) postdoctoral fellowship.

References

Alfaro, L. and Shavlovsky, M. Crowdgrader: A tool for crowdsourcing the evaluation of homework assignments. In *Technical Symposium on Computer Science Education (SIGCSE)*, 2014.

Berend, D. and Kontorovich, A. A finite sample analysis of the naive bayes classifier. *Journal of Machine Learning Research (JMLR)*, 16:1519–1545, 2015.

Bhatia, R. and Davis, C. A better bound on the variance. *The American Mathematical Monthly*, 107(4):353–357, 2010.

Bonald, T. and Combes, R. A minimax optimal algorithm for crowdsourcing. In *Neural Information Processing Systems (NIPS)*, 2017.

Dalvi, N., Dasgupta, A., Kumar, R., and Rastogi, V. Aggregating crowdsourced binary ratings. In *World Wide Web Conference (WWW)*, 2013.

Dawid, A. and Skene, A. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics*, 28(1):20–28, 1979.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Gao, C. and Zhou, D. Minimax optimal convergence rates for estimating ground truth from crowdsourced labels. arXiv:1310.5764 [stat.ML], 2013.

Gao, C., Lu, Y., and Zhou, D. Exact exponent in optimal rates for crowdsourcing. In *International Conference on Machine Learning (ICML)*, 2016.

Ghosh, A., Kale, S., and McAfee, P. Who moderates the moderators? Crowdsourcing abuse detection in user-generated content. In *Conference on Electronic Commerce (EC)*, 2011.

Jaffe, A., Fetaya, E., Nadler, B., Jiang, T., and Kluger, Y. Unsupervised ensemble learning with dependent classifiers. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2016.

Jagabathula, S., Subramanian, L., and Venkataraman, A. Identifying unreliable and adversarial workers in crowdsourced labeling tasks. *Journal of Machine Learning Research*, 18(93):1–67, 2017.

Karger, D., Oh, S., and Shah, D. Iterative learning for reliable crowdsourcing systems. In *Neural Information Processing Systems (NIPS)*, 2011a.

Karger, D., Oh, S., and Shah, D. Budget-optimal crowdsourcing using low-rank matrix approximations. In *Allerton Conference on Communication, Control, and Computing*, 2011b.

Karger, D., Oh, S., and Shah, D. Efficient crowdsourcing for multi-class labeling. In *ACM Sigmetrics*, 2013.

Karger, D., Oh, S., and Shah, D. Budget-optimal task allocation for reliable crowdsourcing systems. *Operations Research*, 65(1):266–287, 2014.

Khetan, A. and Oh, S. Reliable crowdsourcing under the generalized Dawid-Skene model. arXiv:1602.03481v1 [cs.LG], 2016.

Khetan, A., Lipton, Z., and Anandkumar, A. Learning from noisy singly-labeled data. arXiv:1712.04577 [cs.LG], 2017.

Krizhevsky, A. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Le, J., Edmonds, A., Hester, V., and Biewald, L. Ensuring quality in crowdsourced search relevance evaluation: The effects of training question distribution. In *SIGIR workshop on crowdsourcing for search evaluation (CSE)*, 2010.

Li, H., Yu, B., and Zhou, D. Error rate bounds in crowdsourcing models. arXiv:1307.2674 [stat.ML], 2013.

Liao, H., Zeng, A., Xiao, R., Ren, Z.-M., Chen, D.-B., and Zhang, Y.-C. Ranking reputation and quality in online rating systems. *PLoS ONE*, 9(5):e97146, 2014.

Liu, Q., Peng, J., and Ihler, A. Variational inference for crowdsourcing. In *Neural Information Processing Systems (NIPS)*, 2012.

Ma, Y., Saligrama, V., and Szepesvari, C. Crowdsourcing with sparsely interacting workers. arXiv:1706.06660 [cs.LG], 2017.

Nitzan, S. and Paroush, J. Optimal decision rules in uncertain dichotomous choice situations. *International Economic Review*, 23(2):289–297, 1982.

Raykar, V. and Yu, S. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *Journal of Machine Learning Research*, 13:491–518, 2012.

Shah, N., Balakrishnan, S., and Wainwright, M. A permutation-based model for crowd labeling: Optimal estimation and robustness. arXiv:1606.09632 [cs.LG], 2016.

Snow, R., O'Connor, B., Jurafsky, D., and Ng, A. Cheap and fast — but is it good? Evaluating non-expert annotations for natural language tasks. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2008.

Szepesvari, D. A statistical analysis of the aggregation of crowdsourced labels. Master's thesis, University of Waterloo, 2015.

Zhang, Y., Chen, X., Zhou, D., and Jordan, M. Spectral methods meet EM: A provably optimal algorithm for crowdsourcing. *Journal of Machine Learning Research*, 17(102):1–44, 2016. Code available on `https://github.com/zhangyuc/SpectralMethodsMeetEM`.

Zhou, D., Basu, S., Mao, Y., and Platt, J. Learning from the wisdom of crowds by minimax entropy. In *Neural Information Processing Systems (NIPS)*, 2012.