SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

March 2023, Vol. 66 132103:1–132103:17
https://doi.org/10.1007/s11432-020-3067-8

Learning from crowds with robust support vector machines

Wenjun YANG[1], Chaoqun LI[1*] & Liangxiao JIANG[2]

[1]*School of Mathematics and Physics, China University of Geosciences, Wuhan* 430074*, China;*
[2]*School of Computer Science, China University of Geosciences, Wuhan* 430074*, China*

Received 2 May 2020/Accepted 16 July 2020/Published online 8 December 2022

Abstract Crowdsourcing system provides an easy way to obtain labeled training data. However, the labels provided by non-expert labelers often appear low quality. So in practice, each sample usually obtains a multiple label set from multiple different labelers. Learning-from-crowds (LFC) aims to design ground truth inference algorithms to infer the unknown true labels of data from multiple label sets. Despite their proper statistical foundations, the existing ground truth inference algorithms show limited performance when the number of labelers is small. However, more labelers mean higher costs. This paper tries to propose a novel ground truth inference algorithm which can maintain moderate performance and simultaneously reduce labeling costs. This paper addresses LFC from a point of view of robust classifiers and presents a new label noise robust support vector machine inference (RSVMI) algorithm. We prove that only one convex quadratic programming problem needs to be solved to build a robust support vector machine. Furthermore, in order to apply the robust support vector machine to crowdsourced data, two methods are proposed to estimate the noise level of integrated labels. By transforming the original LFC problem into a robust classifier learning problem, our algorithm shows good performance when the number of labelers is very small. In our experiments, the minimum number of labelers is set to 3. In terms of both label quality and model quality, the experimental results on benchmark data sets and real-world data sets show the effectiveness of RSVMI.

Keywords crowdsourcing learning, ground truth inference, integrated labels, label noise, robust support vector machines

Citation Yang W J, Li C Q, Jiang L X. Learning from crowds with robust support vector machines. Sci China Inf Sci, 2023, 66(3): 132103, https://doi.org/10.1007/s11432-020-3067-8

1 Introduction

The research of machine learning usually needs a lot of labeled data, and the quantity and quality of labeled data will directly affect the performance of machine learning algorithms [1, 2]. However, it is usually expensive and time-consuming to acquire the true labels of data from domain experts. With the development of crowdsourcing platforms, it has become cheaper and faster to collect labels of data by employing ordinary crowd workers (web labelers).

Unfortunately, a single non-expert labeler may provide incorrect labels. It may be caused by personal preference, low payment for each task, and varying cognitive abilities. To solve this issue, it is usually recommended to label every task multiple times by different workers (i.e., repeated labeling) [3], and so every sample has a multiple label set provided by repeating labeling. For example, for a sample i described by a d-dimensional feature vector, y is the corresponding unknown true label. J crowd workers are employed to label the sample, and the resulting multiple label set is $\boldsymbol{l}_i = \{l_i^j\}_{j=1}^J$, where l_i^j is the label of the sample i annotated by the jth labeler. Therefore, it is crucial to design a suitable ground truth inference algorithm to infer the label of every sample from its multiple label set. The label $\hat{y}$ obtained by an inference algorithm is called the integrated label of the sample. In the past few years, ground truth inference algorithms have attracted a lot of attention.

Majority voting (MV) is the simplest and most straightforward ground truth inference algorithm. In MV, the label which obtains the maximum number of votes in the multiple label set is treated as the

* Corresponding author (email: chqli@cug.edu.cn)

integrated label of a sample. MV ignores some factors, such as the qualities of labelers, and the difficulty of labeling each task correctly. If we use MV directly, the generated label is rough and biased. Therefore, many scholars have proposed various algorithms to make fuller use of the information provided by labelers. These algorithms can be generally divided into three categories. (1) Weighted majority voting (WMV). This type of algorithms include MV-Freq, MV-Beta [4], IWMV [5], M^{3}V [6], DEWMV [7], MNLDP [8], WMV-Freq, and WPaired-Freq [9]. (2) Algorithms based on expectation maximum (EM). These algorithms include DS [10], RY [11] and ZenCrowd (ZC) [12], PLAT [13], and RGTIA [14]. (3) Other algorithms, such as KOS [15], GTIC [16], and VGPCR [17]. Although all these algorithms show good performance in practice, we notice that generally the performance of these algorithms will present the decline of different degrees if the number of labelers is small. The main reason is that the smaller the number of labelers is, the greater the uncertainty of multiple label sets is. For example, many ground truth inference algorithms need to estimate empirical probability or posteriori probability. When there are only a few crowdsourced labels available for each sample, the effectiveness of probability estimation is heavily affected. The great uncertainty of multiple label sets severely weakens the performance of ground truth inference algorithms.

So for many existing ground inference algorithms, their performance is usually limited if there are not sufficient labelers. However, more labelers mean more costs. Therefore, in order to maintain moderate performance and reduce the labeling costs (i.e., the number of labelers) simultaneously, this paper tries to propose a novel ground truth inference algorithm.

Our work infers integrated labels from a point of view of robust classifiers. Empirical study has proved that integrated labels still contain a percentage of noise, no matter which ground truth inference algorithm is used to infer the integrated labels [18]. Here noise refers to the samples, the integrated labels of which are different from their true labels. As mentioned above, the smaller the number of labelers is, the greater the uncertainty of multiple label sets is. Then the integrated labels inferred from the multiple label sets may contain more noise. So the key of ground truth inference is how to cope with label noise. Motivated by this idea, this paper proposes a novel ground truth inference algorithm based on robust support vector machines (SVM).

In this algorithm, we first use MV to infer the integrated labels of training samples and form an inferred data set D^N. Obviously, the integrated labels of these samples do not exactly match their true labels. Then, we estimate the noise level of integrated labels (i.e., the probability that each sample's integrated label does not match its true label). By embedding the noise level into the building process of support vector machine, we build a robust SVM. Finally, we use the robust SVM to update the integrated labels of the data set D^N. Thus we call our algorithm robust support vector machine inference (RSVMI).

In order to implement RSVMI, there are two problems to be solved. (1) How to build a trainable robust SVM. (2) How to estimate the noise level of integrated labels. For the first problem, we use the label noise level to modify the optimization target of SVM, so that the classifier would not overfit the noise labels. What is more, we prove that the modified optimization problem is still a convex optimization problem. For the second problem, when we apply the robust SVM to crowdsourced data, the noise level of integrated labels has to be estimated. This paper proposes two methods to estimate the noise level of integrated labels based on a binomial distribution and a modified sigmoid function respectively. Because RSVMI builds an SVM which is robust to label noise, when the number of labelers is small and the uncertainty of multiple label sets is great, RSVMI still maintains moderate performance. Moreover, although our algorithm modifies the optimization function of SVM, we prove that it is still a convex optimization problem. However, most of the existing expectation-maximum (EM)-style inference methods model the true labels as latent variables, and the resulting optimization problems are not convex. Compared with these methods, RSVMI is guaranteed to get the optimal solution.

In summary, the contributions of this paper are as follows.

(1) We propose a modified SVM which is robust to label noise, and prove that the optimization problem of the robust SVM is still a convex quadratic programming problem.

(2) In order to apply the robust SVM to crowdsourced data, two methods are proposed to estimate the noise level of integrated labels.

(3) Our algorithm greatly reduces the labeling costs and at the same time maintains moderate performance.

The rest of this paper is organized as follows. Section 2 introduces the related work on ground truth inference algorithms for crowdsourcing. Section 3 describes our algorithm RSVMI. In Section 4, we report the experimental results on the benchmark data sets and the real-world crowdsourced data sets.

In Section 5, we conclude the paper and outline the main directions for future work.

2 Related work

Using repeated labeling to obtain multiple labels is practiced in applications where labeling is not perfect. However, this approach introduces some scientifically challenging issues, such as combining the unknown expertise of labelers and dealing with disagreements over labeled samples. To address these issues, there have been a lot of studies on how to infer the integrated labels from the multiple labels over the past few years.

MV is the simplest and most straightforward algorithm, but it is not reasonable that MV assumes that all labels are equally good. Obviously, the labels provided by reliable annotators should have higher credibility, and so these labels should have greater weights. Therefore, WMV is used to improve MV. Tian and Zhu [6] proposed a new concept-crowdsourcing margin which transforms the ground truth inference problem into an optimization problem. The reliability of each labeler is obtained by solving the optimization problem. Recently, the work in [7] proposed an algorithm based on differential evolution to solve the label integration problem. By setting up fitness functions for weights, differential evolution algorithms are used to calculate the weight of each crowdsourced label. Zhang et al. [8] proposed a ground truth inference algorithm named MNLDP. MNLDP is based on an assumption that it is more likely that two samples with a small distance have the same true label. In MNLDP, each sample absorbs a part of the multi-noise label distribution from its nearest neighbors, but at the same time maintains a part of its own multi-noise label distribution.

Besides WMV, there are many ground truth inference algorithms based on EM, such as DS [10], GLAD [19], RY [11] and ZC [12]. EM-based algorithms tend to use probability parameters to measure the reliabilities of labelers and (or) the difficulty of labeling samples, and the EM algorithms are used to calculate these probability parameters and hidden true labels. For example, DS established a confusion matrix π for each labeler in the inferring process. The element π_{ij} in the confusion matrix denotes the probability that the worker provides the label i to the sample with the true label j. Similarly, RY proposed two parameters sensitivity and specificity. In RY, the parameter sensitivity represents the labeler's prejudice to the positive label and the parameter specificity represents the labeler's prejudice to the negative label.

In recent years, some scholars have proposed different ground truth inference algorithms. Karger et al. [15] proposed a new algorithm KOS based on belief propagation and low-rank matrix approximation for deciding which tasks to assign to which workers and for inferring true labels from the crowdsourced labels. Rodrigues et al. [20] introduced a crowdsourced classifier based on Gaussian processes (GP). In their model, the true underlying labels are treated as latent variables by means of a GP. On the basis of [11, 20], Ruiz et al. [17] proposed an algorithm to infer all unknowns by Variation Bayes. In addition, one future direction is how to adapt partial label learning algorithms [21] and deep learning algorithms [22, 23] to utilize the information provided by crowd labelers to overcome the negative effects of the labeling uncertainty.

Although researchers have done a lot of studies on ground truth inference, for many algorithms such as WMV and algorithms based on EM, their performance is closely related to the number of labelers. More crowdsourced labels are collected for each sample, better performance can these algorithms show. But more crowdsourced labels mean higher costs. Thus, in this paper, we propose an inference algorithm based on a label noise robust SVM. With the help of the label noise robust SVM, our method still shows good performance when the number of labelers is small.

3 Robust support vector machine inference model

3.1 A label noise robust SVM for uniform noise

Building a robust classifier is not a new technique in the machine learning community, especially in adversary learning [24]. Biggio et al. [25] proposed a preliminary investigation of the robustness of SVM against adversarial data manipulation. In their paper, they assumed that the adversary controls some training data in order to disrupt the SVM learning process. In other words, there are some adversaries to deliberately flip the labels of some training samples (i.e., the positive (negative) samples are flipped to the

negative (positive)), so that the classifier is misled. Since it is impossible to predict which samples would be flipped in advance, they assumed that the probability of each sample being flipped was μ. Based on this assumption, for a binary classification problem, label noise can be explicitly modeled by assuming that the labels in the training data set $\{\boldsymbol{x}_i, y_i\}_{i=1}^n \in \mathcal{X} \times \{-1, +1\}$ can be flipped and the probability of being flipped is μ.

For a noise-free data set, a soft-margin SVM [26, 27] usually needs to solve the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{w},b,\xi} \quad & \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + C\sum_{i=1}^{n}\xi_i \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) \geqslant 1 - \xi_i, i = 1, \ldots, n, \\ & \xi_i \geqslant 0, i = 1, \ldots, n. \end{aligned} \tag{1}$$

Where the variable ξ_i denotes the extent to which the sample $\boldsymbol{x}_i$ violates the margin and n is the number of training samples. $\boldsymbol{w}, b$ are the parameters of the decision hyperplane. In order to solve the problem (1) effectively, it is usually transformed into a dual problem by Lagrange multiplier method. In matrix form, this dual problem can be written as

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}_n^{\mathrm{T}}\boldsymbol{\alpha} \\ \text{s.t.} \quad & 0 \leqslant \alpha_i \leqslant C,\ i = 1, \ldots, n, \\ & \sum_{i=1}^{n}\alpha_i y_i = 0. \end{aligned} \tag{2}$$

Where the matrix $\boldsymbol{Q} = \boldsymbol{K} \circ \boldsymbol{y}\boldsymbol{y}^{\mathrm{T}}$, $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]^{\mathrm{T}}$, and $\mathbf{1}_n$ is a column vector of n ones. The elements of matrix $\boldsymbol{Q}$ are $Q_{ij} = y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j), i, j = 1, 2, \ldots, n$. $\boldsymbol{K}$ is the kernel matrix, whose elements $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^{\mathrm{T}}\phi(\boldsymbol{x}_j), i, j = 1, 2, \ldots, n$. The non-linear map $\phi : \mathcal{X} \to \Phi$ maps training samples to a higher dimensional feature space.

Owing to the label noise in the training data set, each label y_i is replaced by $y_i' = y_i(1 - 2\epsilon_i)$, where the variable ϵ_i represents whether the sample's label y_i is flipped ($\epsilon_i = 1$) or not ($\epsilon_i = 0$). Obviously, in the dual problem, the label noise will affect the matrix $\boldsymbol{Q}$ in the optimization problem (2). Thus, the elements of matrix $\boldsymbol{Q}$ are written as

$$Q_{ij} = y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j)(1 - 2\epsilon_i)(1 - 2\epsilon_j), \quad i, j = 1, 2, \ldots, n. \tag{3}$$

As mentioned above, Biggio et al. [25] assumed each label y_i is independently flipped with the probability μ. In other words, $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$ are n i.i.d. boolean random variables, and μ is the probability of $\epsilon_i = 1$. The variance of ϵ_i is $\sigma^2 = \mu(1 - \mu)$. Under this assumption, they used the expected value of Q_{ij} to replace Q_{ij}.

$$\mathrm{E}\left[Q_{ij}\right] = \begin{cases} y_i y_j K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)\left(1 - 4\sigma^2\right), & \text{if } i \neq j, \\ y_i y_j K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right), & \text{otherwise.} \end{cases} \tag{4}$$

The final optimization problem is written as

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^{\mathrm{T}}\boldsymbol{\alpha} + \frac{S}{1 - S}\left[\frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\left(\boldsymbol{Q} \circ \mathbb{I}_{n\times n}\right)\boldsymbol{\alpha} - \mathbf{1}^{\mathrm{T}}\boldsymbol{\alpha}\right] \\ \text{s.t.} \quad & 0 \leqslant \alpha_i \leqslant C,\ i = 1, \ldots, n, \\ & \sum_{i=1}^{n}\alpha_i y_i = 0, \end{aligned} \tag{5}$$

where $S = 4\sigma^2$, and $\mathbb{I}_{n\times n}$ is a unit matrix. Obviously, the robust SVM based on (5) is more suitable for uniform noise case, because it assumes that the probability of label flipping is the same for each sample.

3.2 A label noise robust SVM for crowdsourcing

We must admit that no matter which inference algorithm is used, there is always some noise in the integrated labels. Thus, the key of crowdsourcing learning is how to cope with label noise. To this end, we try to use a robust SVM to infer true labels.

However, the robust SVM proposed by [25] is more suitable for uniform noise case, and such a model is not suitable for crowdsourcing system because this model ignores the different abilities of labelers and the difficulties of samples.

Here we assume that the random variables $\epsilon_1, \epsilon_2, \ldots, \epsilon_n$ are independent of each other, and they all obey two-point distributions. q_i is the probability of $\epsilon_i = 1$. Thus the expected value of Q_{ij} is written as

$$\mathrm{E}\left[Q_{ij}\right] = \begin{cases} y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j)(1-2q_i)(1-2q_j), & \text{if } i \neq j, \\ y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j), & \text{otherwise.} \end{cases} \tag{6}$$

Now, we can use the expected value of Q_{ij} to replace Q_{ij}. Therefore, we only need to solve the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{E}[\boldsymbol{Q}]\boldsymbol{\alpha} - \mathbf{1}_n^{\mathrm{T}}\boldsymbol{\alpha} \\ \text{s.t.} \quad & 0 \leqslant \alpha_i \leqslant C, \ i = 1, \ldots, n, \\ & \sum_{i=1}^{n} \alpha_i y_i = 0, \end{aligned} \tag{7}$$

where the elements of matrix $\boldsymbol{E}[\boldsymbol{Q}]$ are $\mathrm{E}[Q_{ij}]$ $(i, j = 1, 2, \ldots, n)$ defined in (6). Although our hypothesis makes the model unable to be simplified to the form similar to (5), it does not make the optimization problem difficult to solve. The proposed method only yields a kernel matrix correction. It is easy to prove that as long as the original kernel K is positive definite, the problem (7) is still a convex quadratic programming problem and is guaranteed to get the optimal solution. Although the original optimization problem has been modified, it does not make the problem too complicated.

Theorem 1. If the original kernel K is positive definite, the problem (7) is still a convex quadratic programming problem.

Proof. Since K is a positive definite kernel on $\mathcal{X} \times \mathcal{X}$, there is a map ϕ, which makes

$$K(\boldsymbol{x}, \boldsymbol{z}) = \phi(\boldsymbol{x}) \cdot \phi(\boldsymbol{z}),$$

so for any vector $\boldsymbol{c} = (c_1, c_2, \ldots, c_n)^{\mathrm{T}}$,

$$\begin{aligned} & \sum_{i,j=1}^{n} c_i c_j (1-2q_i)(1-2q_j)(\phi(\boldsymbol{x}_i) \cdot \phi(\boldsymbol{x}_j)) y_i y_j \\ & = \left(\sum_{i=1}^{n} c_i (1-2q_i) \phi\left(x_i\right) y_i \right) \cdot \left(\sum_{j=1}^{n} c_j (1-2q_j) \phi\left(x_j\right) y_j \right) \\ & = \left\| \sum_{i=1}^{n} c_i (1-2q_i) \phi\left(x_i\right) y_i \right\|^2 \geqslant 0. \end{aligned} \tag{8}$$

Owing to $q_i \in [0, 1]$, $4q_i(1-q_i) \geqslant 0$, then

$$\sum_{i=1}^{n} c_i^2 4q_i(1-q_i)\phi(x_i)^2 y_i^2 \geqslant 0. \tag{9}$$

Thus,

$$\begin{aligned} \boldsymbol{c}^{\mathrm{T}} \boldsymbol{E}[\boldsymbol{Q}]\boldsymbol{c} = & \sum_{i,j=1}^{n} c_i c_j (1-2q_i)(1-2q_j)(\phi(\boldsymbol{x}_i) \cdot \phi(\boldsymbol{x}_j)) y_i y_j \\ & + \sum_{i=1}^{n} c_i^2 4(q_i - q_i^2) y_i^2 \phi(x_i)^2 \geqslant 0. \end{aligned} \tag{10}$$

Therefore, $\boldsymbol{E}[\boldsymbol{Q}]$ is still a positive semidefinite matrix, and the problem (7) is still a convex quadratic optimization problem.

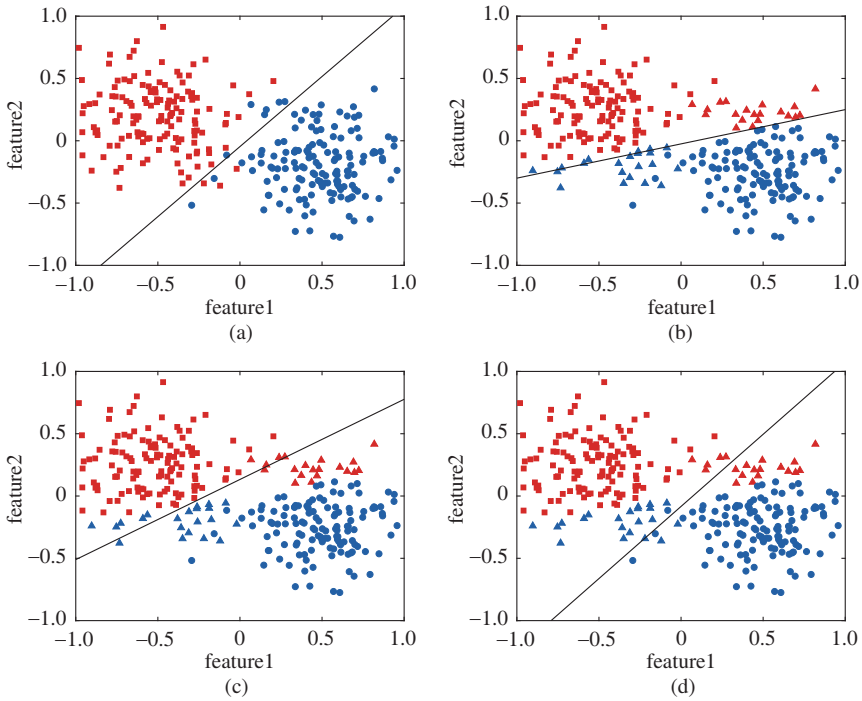

Figure 1 (Color online) Standard SVMs trained on (a) untainted and (b) tainted data (the first and second plots, respectively), robust SVMs with (c) $\mu = 0.16$ and (b) different q_i trained on tainted data (the third and fourth plots). The lines in the middle of the data clouds represent the classification boundaries.

Now, let us take an artificial binary toy classification data set as an example to illustrate the effectiveness of the robust SVMs. The data set is artificially generated from two normal distributions. It contains 600 data points with two-dimensional features, half for training and half for testing. In this experiment, we divide the data set into three parts. The two features of each sample in the first part are both greater than 0, the two features of each sample in the second part are both less than 0, and the remaining third part. We only flip the samples of the first two parts, and the probabilities that the samples of these two parts are flipped are 0.85 and 0.75 respectively. On the basis of this setting, we randomly flip the labels of 49 training samples. For the robust SVM based on (5), $\mu = 49/300 \approx 0.16$. For our robust SVM based on (7), the corresponding q_i of the samples of the first and second parts are 0.85 and 0.75 respectively, and the corresponding q_i of the samples of the third part is 0.

Figure 1 graphically shows the detailed process and results. In Figure 1, each red square (blue dot) represents a positive (negative) sample. The symbol of the triangle means that this sample has been flipped, that is, the red (blue) triangle means that the correct class of the sample is negative (positive), but is flipped as a positive (negative) sample.

Figure 1(a) shows the original SVM trained on untainted data, with an accuracy of 98.33% on the test set. Figure 1(b) shows the original SVM trained on tainted data, with an accuracy of 89.33% on the test set. Figure 1(c) shows the robust SVM based on (5) ($\mu = 0.16$) trained on tainted data, with an accuracy of 92.67% on the test set. Figure 1(d) shows our robust SVM based on (7) (different q_i) trained on tainted data, with an accuracy of 95.33% on the test set. Obviously, label noise significantly weakens the SVM's performance. Both robust SVMs can counteract the label noise to some extent. As we expected, in the case that label noise is not completely random, our robust SVM based on (7) performs better. In this experiment, the setting mode of q_i (only three values) is a little rough. In order to use our robust SVM to infer integrated labels of crowdsourced data, the key is how to estimate $q_i, i = 1, 2, \ldots, n$. Next, we propose two methods to estimate these probabilities.

3.3 Ground truth inference using robust SVMs

As mentioned above, in order to apply our robust SVM to solve ground truth inference problems, we need to accurately estimate the probability that the label of each sample i is flipped, q_i. In this paper, we propose two methods to estimate q_i $(i = 1, 2, \ldots, n)$, where n is the number of the samples. It is

worth noting that in Subsection 3.2 we propose an assumption that each label is flipped independently. Generally, workers in the crowdsourcing system make decisions independently. We can approximately think that in a crowdsourcing system, the probability of each integrated label being flipped is also independent of each other.

First, we reconsider the mathematical form of MV in binary classification tasks. For a sample i, it associates a multiple label set $\boldsymbol{l}_i = \{l_i^j\}_{j=1}^J$, where J is the number of workers and $l_i^j \in \{+1, 0, -1\}$, $+1$, -1, 0 represent positive, negative, and null (i.e., labeler j did not label the sample i) respectively. Let the number of non-zero elements in the set $\boldsymbol{l}_i$ be N_i.

For the sample i, MV assigns it an estimated label $\hat{y}_i$ which is regarded as a true label temporarily. On this basis, the quality of the labeler j is defined as

$$p_j = \frac{\sum_{i=1}^n \delta(l_i^j = \hat{y}_i)}{\sum_{i=1}^n |l_i^j|}, \tag{11}$$

where $\delta(\cdot)$ is the indicator function. Thus, we define the average labeling quality of the workers who label the sample i as

$$\overline{p}_i = \frac{\sum_{j=1}^J p_j |l_i^j|}{N_i}. \tag{12}$$

As long as the number of positive labels in the multiple label set $\boldsymbol{l}_i$ exceeds $N_i/2$, the integrated label will be positive. So the probability of the integrated label being flipped obeys a binomial distribution, and q_i corresponding to each sample i is

$$q_i = \Pr(\hat{y}_i \neq y_i) = \sum_{m=0}^{\lfloor N_i/2 \rfloor} \binom{N_i}{m} \overline{p}_i^m (1 - \overline{p}_i)^{N_i - m}. \tag{13}$$

Using (13), we can get the probability that each sample's integrated label is flipped. The first approach is based on the idea that label noise comes from false labels provided by labelers.

The second approach is based on the idea that different multiple label sets carry different information. For example, one sample with the multiple label set $\{+, +, +, +, +, +, -\}$ and another sample with the multiple label set $\{+, +, +, +, -, -, -\}$, obviously, in MV, the integrated labels of both samples are positive. However, the probabilities that their integrated labels are flipped are quite different. Intuitively, we think the latter is more likely to be flipped than the former. Based on this idea, we propose another method to estimate the probabilities $q_i, i = 1, 2, \ldots, n$.

We firstly need a scalar to measure the uncertainty of each multiple label set. In information theory and statistics, entropy is usually used to be the uncertainty measure of random variables. The greater the entropy is, the greater the uncertainty of random variables is. In a crowdsourcing system, for a sample i, the entropy of its multiple label set $\boldsymbol{l}_i$ is defined as

$$H(\boldsymbol{l}_i) = -p_{\text{neg}} \log_2(p_{\text{neg}}) - p_{\text{pos}} \log_2(p_{\text{pos}}), \tag{14}$$

where

$$p_{\text{neg}} = \frac{\text{Neg}}{\text{Pos} + \text{Neg}}, \quad p_{\text{pos}} = \frac{\text{Pos}}{\text{Pos} + \text{Neg}}, \tag{15}$$

Pos and Neg are the numbers of positive and negative samples respectively. Because entropy does not directly reflect the probability of label flipping, we use a modified sigmoid function to convert it into probability

$$q_i = \Pr(\hat{y}_i \neq y_i) = 2\left(\frac{1}{1 + \mathrm{e}^{-tH(\boldsymbol{l}_i)}} - \frac{1}{2}\right), \tag{16}$$

where $t > 0$ is a hyperparameter that adjusts the probability. When t is fixed, q_i is an increasing function of $H(\boldsymbol{l}_i)$. In (16), when the entropy $H(\boldsymbol{l}_i) = 0$, the probability $q_i = 0$. When the entropy $H(\boldsymbol{l}_i)$ reaches the maximum value of 1, the probability $q_i(t) = 2(\frac{1}{1+\mathrm{e}^{-t}} - \frac{1}{2})$. Obviously, $q_i(t)$ is a monotonically increasing

function, where $q_i(1) = 0.46$, $q_i(2) = 0.76$, $q_i(3) = 0.91$, $q_i(4) = 0.96$ and $q_i(5) = 0.98$. When $t \geqslant 4$, $q_i(t)$ is already close to 1 and the increasing trend slows down. Thus, in our paper, we set $t = 4$.

Now we summarize our inference algorithm as follows. When we get a training data with crowdsourced labels, firstly we use MV to get the initial integrated labels of data, and estimate the noise level of integrated labels, that is, we calculate $q_i, i = 1, 2, \ldots, n$ using (13) or (16). Then we plug these probabilities into the problem (7) and the robust SVM based on (7) is built. Finally the robust SVM is used to update the integrated labels of data. We call our algorithm robust support vector machine inference (RSVMI). Our algorithms based on (13) and (16) are denoted as RSVMI$^{\mathrm{w}}$ and RSVMI$^{\mathrm{e}}$ respectively. Algorithm 1 describes the detailed algorithmic procedure of RSVMI.

Algorithm 1 RSVMI

Require: Data set $\{\boldsymbol{x}_i\}_{i=1}^n$ and the corresponding multiple label set $\{\boldsymbol{l}_i\}_{i=1}^n$.
Ensure: Integrated labels $\{\hat{y}_i\}_{i=1}^n$.
1: Use MV to get the initial integrated labels of $\{\boldsymbol{x}_i\}_{i=1}^n$.
2: Use (13) or (16) to estimate probabilities $q_i, i = 1, 2, \ldots, n$.
3: Build the robust SVM based on (7).
4: Use the robust SVM to relabel data set $\{\boldsymbol{x}_i\}_{i=1}^n$, and get the updated integrated labels $\{\hat{y}_i\}_{i=1}^n$.
5: Return $\{\hat{y}_i\}_{i=1}^n$.

4 Experiments and results

4.1 Experiments on benchmark data sets

In all these experiments, we compare our ground truth inference algorithms RSVMI$^{\mathrm{e}}$ and RSVMI$^{\mathrm{w}}$ with MV, DS, KOS, RY and MNLDP. The performance of these algorithms is compared using two general metrics: the label quality of data and the model quality of built target classifier (i.e., classification accuracy). SVM is chosen as the target classifier. For both the robust classifier and the target classifier, we use linear kernel. The parameter t in (16) is set to 4. We implement RSVMI and MNLDP on the CEKA platform [28] and use the existing implementations of MV, RY, DS and KOS on the CEKA platform. We also use the existing implementation of SVM on the WEKA platform [29]. All experiment results are obtained via 10 runs of 5-fold-cross-validation.

In this part, each ground truth inference algorithm will be tested on 18 binary classification benchmark data sets from the UCI machine learning databases [30], which represent a wide range of domains and data characteristics. In order to obtain the simulated multiple label sets of data, each sample is labeled by J simulated labelers. It is worth noting that in the benchmark data experiments, each virtual labeler labels all training samples, so the number of labelers J represents that each sample gets J crowdsourced labels. The label quality of each labeler is p_j $(j = 1, 2, \ldots, J)$. That is, the jth labeler will assign a sample to its true class with probability p_j and the opposite value with probability $1 - p_j$. The labeling quality of each labeler p_j is generated randomly from a uniform distribution on the interval $[0.55, 0.75]$, i.e., $p_j \in [0.55, 0.75]$ $(j = 1, 2, \ldots, J)$. After each sample obtains the simulated multiple label set, we apply seven ground truth inference algorithms: RSVMI$^{\mathrm{w}}$, RSVMI$^{\mathrm{e}}$, MV, KOS, RY, DS and MNLDP to infer integrated labels. The test set does not involve the calculation of label quality.

In order to compare the impact of the number of labelers on the performance of inference algorithms, two series of experiments were run:

- In the first series of experiments, the performance of seven algorithms on two randomly selected data sets is compared as the number of labelers increases from 3 to 29.
- In the second series of experiments, the number of labelers is fixed at 3 and 5, and the performance of 7 algorithms on 18 data sets is compared.

4.1.1 *The first series of experiments*

In order to observe the relationship between the performance of these ground truth inference algorithms and the number of labelers, we made some explored experiments on two randomly selected data sets breast-w and diagnosis. The above seven algorithms were run on the two data sets, and the number of labelers is from 3 to 29. Figure 2 shows the changing trend of the label qualities with the number of labelers. From Figure 2, we can see that both RSVMI$^{\mathrm{w}}$ and RSVMI$^{\mathrm{e}}$ show better performance compared

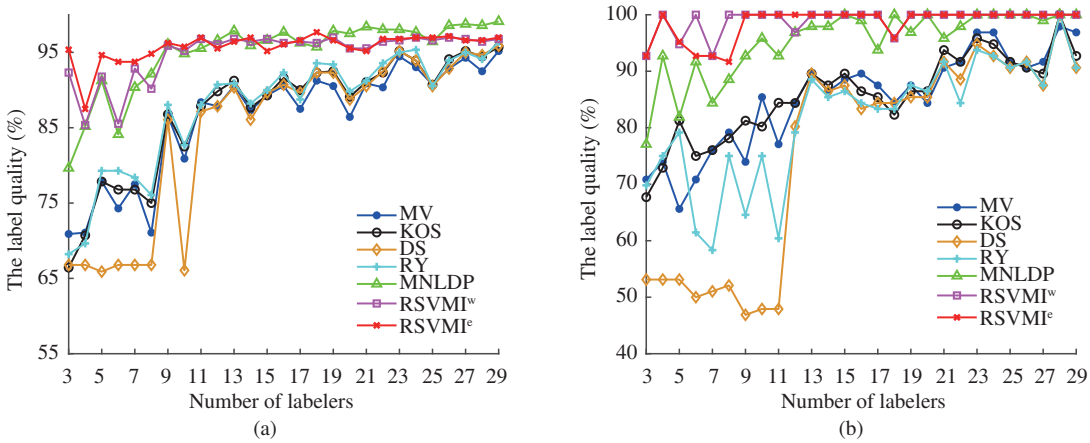

Figure 2 (Color online) Comparisons of label quality under different numbers of labelers. (a) Breast-w data set; (b) diagnosis data set.

Table 1 The label quality (%) comparisons with $J = 3$

Data set	MV	KOS	DS	RY	MNLDP	RSVMI^{w}	RSVMI^{e}
blood	74.06	74.06	76.20	74.36	74.33	76.30	76.54
breast-cancer	67.85	46.53	70.30	66.79	70.64	71.34	73.10
breast-w	65.95	51.59	65.52	65.20	77.76	84.93	87.40
credit-a	63.19	54.17	51.96	66.38	71.96	78.88	80.04
diagnosis	69.17	59.58	50.42	66.04	78.54	87.08	88.33
haberman	72.89	62.71	73.53	72.56	72.55	72.89	73.21
heart-c	68.65	67.09	52.07	66.34	75.17	79.04	81.02
heart-h	64.96	55.40	63.95	63.84	71.43	76.18	79.76
heart-statlog	71.11	59.63	55.56	67.41	78.80	82.13	82.22
hepatitis	70.97	57.26	79.35	67.42	73.39	79.52	81.45
house-vote	59.31	58.62	61.38	57.47	66.67	71.44	82.01
income	64.17	42.13	51.25	65.75	67.25	73.71	75.88
ionosphere	68.38	54.00	64.10	68.52	71.15	84.76	78.28
labor	70.16	39.84	64.88	52.17	82.91	85.08	89.03
pima	62.76	50.64	65.10	63.87	67.51	71.09	71.16
splice	71.61	71.61	51.82	70.31	84.64	95.70	97.14
vote	57.93	53.45	51.84	55.52	69.89	74.60	88.56
z-alizadeh-sani	69.97	49.97	71.28	70.13	71.70	74.42	78.80
Average	67.39	56.02	62.25	65.56	73.68	78.84	81.33
Average ranking	4.6944	6.4444	5.1111	5.2778	3.3333	2.0278	1.1111

with the other five algorithms. Especially when the number of labelers is relatively small, the label qualities of RSVMI^{w} and RSVMI^{e} are much higher than other five algorithms. And of course, the gap among these algorithms is gradually closing as the number of labelers increases. This also confirms what we said before, for many algorithms, their performance is closely related to the number of crowdsourced labels. When the number of crowdsourced labels is small, their performance is limited. But as the number of crowdsourced labels increases, their performance has significant improvement.

4.1.2 *The second series of experiments*

In order to further test the performance of our algorithms, in the second series of experiments, all seven ground truth inference algorithms were run on 18 data sets and the number of labelers was fixed at 3 and 5. Tables 1–4 show the experimental results when the number of labelers is 3. Tables 5–8 show the experimental results when the number of labelers is 5.

Tables 1, 2, 5 and 6 show the detailed integrated label qualities and model qualities. For the further comparison of multiple algorithms over multiple data sets [31, 32], we employ the KEEL data-mining software tool [33] to conduct a Friedman test with the corresponding post-hoc tests [34,35] (e.g., Nemenyi

Table 2 The classification accuracy (%) comparisons with $J = 3$

Data set	MV	KOS	DS	RY	MNLDP	RSVMI^{w}	RSVMI^{e}
blood	76.18	76.18	76.18	76.18	76.18	76.45	76.45
breast-cancer	68.87	68.87	70.28	67.82	72.02	69.27	72.07
breast-w	95.57	95.57	65.54	94.30	97.00	78.40	79.68
credit-a	73.04	52.75	47.68	71.59	76.67	77.83	78.99
diagnosis	77.50	59.17	40.83	71.67	80.00	90.00	91.67
haberman	72.89	62.71	73.53	72.56	72.55	72.89	73.21
heart-c	74.92	74.92	55.10	71.43	76.54	78.59	79.24
heart-h	76.47	60.76	64.03	63.73	76.17	77.16	81.56
heart-statlog	75.56	75.19	55.56	74.81	77.78	78.78	80.74
hepatitis	72.26	52.90	79.35	76.13	62.58	76.13	78.06
house-vote	81.84	77.01	61.38	72.18	74.94	74.94	83.68
income	71.83	71.83	47.00	68.33	71.67	74.67	76.50
ionosphere	72.87	50.77	65.31	54.41	72.87	77.48	92.73
labor	84.20	37.62	64.48	37.06	82.38	84.20	85.73
pima	62.76	50.64	65.10	63.87	67.51	71.09	71.16
splice	73.66	73.66	48.24	74.19	79.66	93.09	95.44
vote	88.51	58.39	61.38	79.54	82.99	87.82	88.28
z-alizadeh-sani	72.62	58.71	71.32	72.56	75.57	73.86	76.89
Average	76.20	64.32	61.79	70.13	76.39	78.48	81.23
Average ranking	3.8333	5.6389	5.3889	5.3611	3.6111	2.7500	1.4167

Table 3 The label quality (%) post-hoc comparisons with $J = 3$ [a)]

i	Algorithm	$z = (R_0 - R_i)/SE$	p
1	KOS vs. RSVMI^{e}	7.406561	0
2	KOS vs. RSVMI^{w}	6.133558	0
3	RY vs. RSVMI^{e}	5.786376	0
4	DS vs. RSVMI^{e}	5.554921	0
5	MV vs. RSVMI^{e}	4.976283	0.000001
6	RY vs. RSVMI^{w}	4.513373	0.000006
7	KOS vs. MNLDP	4.320494	0.000016
8	DS vs. RSVMI^{w}	4.281918	0.000019
9	MV vs. RSVMI^{w}	3.70328	0.000213
10	MNLDP vs. RSVMI^{e}	3.086067	0.002028
11	RY vs. MNLDP	2.700309	0.006928
12	DS vs. MNLDP	2.468854	0.013555
13	MV vs. KOS	2.430278	0.015087
14	MV vs. MNLDP	1.890216	0.058729
15	KOS vs. DS	1.85164	0.064078
16	MNLDP vs. RSVMI^{w}	1.813064	0.069822
17	KOS vs. RY	1.620185	0.105193
18	RSVMI^{w} vs. RSVMI^{e}	1.273003	0.203017
19	MV vs. RY	0.810093	0.417887
20	MV vs. DS	0.578638	0.562834
21	DS vs. RY	0.231455	0.816961

a) Nemenyi's procedure rejects those hypotheses that have an unadjusted p-value $\leqslant 0.002381$:

- KOS vs. RSVMI^{e}
- KOS vs. RSVMI^{w}
- RY vs. RSVMI^{e}
- DS vs. RSVMI^{e}
- MV vs. RSVMI^{e}
- RY vs. RSVMI^{w}
- DS vs. RSVMI^{w}
- MV vs. RSVMI^{w}
- MNLDP vs. RSVMI^{e}

test). The average rankings of algorithms are shown on the bottom of these tables. To avoid redundancy, the calculation processes of Friedman tests will not be listed in detail. Tabels 3, 4, 7 and 8 show the results of these post-hoc tests.

Those results strongly demonstrate the superiority of our algorithm in improving label quality and

Table 4 The classification accuracy (%) post-hoc comparisons with $J = 3$ [a)]

i	Algorithm	$z = (R_0 - R_i)/SE$	p
1	KOS vs. RSVMI^{e}	5.863527	0
2	DS vs. RSVMI^{e}	5.516345	0
3	RY vs. RSVMI^{e}	5.477769	0
4	KOS vs. RSVMI^{w}	4.011887	0.00006
5	DS vs. RSVMI^{w}	3.664705	0.000248
6	RY vs. RSVMI^{w}	3.626129	0.000288
7	MV vs. RSVMI^{e}	3.356098	0.000791
8	MNLDP vs. RSVMI^{e}	3.047491	0.002308
9	KOS vs. MNLDP	2.816036	0.004862
10	MV vs. KOS	2.507429	0.012161
11	DS vs. MNLDP	2.468854	0.013555
12	RY vs. MNLDP	2.430278	0.015087
13	MV vs. DS	2.160247	0.030754
14	MV vs. RY	2.121671	0.033865
15	RSVMI^{w} vs. RSVMI^{e}	1.85164	0.064078
16	MV vs. RSVMI^{w}	1.504458	0.132464
17	MNLDP vs. RSVMI^{w}	1.195851	0.231755
18	KOS vs. RY	0.385758	0.699676
19	KOS vs. DS	0.347183	0.728454
20	MV vs. MNLDP	0.308607	0.757621
21	DS vs. RY	0.038576	0.969229

a) Nemenyi's procedure rejects those hypotheses that have an unadjusted p-value $\leqslant 0.002381$:

- KOS vs. RSVMI^{e}
- DS vs. RSVMI^{e}
- RY vs. RSVMI^{e}
- KOS vs. RSVMI^{w}
- DS vs. RSVMI^{w}
- RY vs. RSVMI^{w}
- MV vs. RSVMI^{e}
- MNLDP vs. RSVMI^{e}

Table 5 The label quality (%) comparisons with $J = 5$

Data set	MV	KOS	DS	RY	MNLDP	RSVMI^{w}	RSVMI^{e}
blood	75.43	75.33	76.20	76.47	76.23	74.50	74.77
breast-cancer	66.06	65.03	70.27	64.31	65.04	62.92	66.42
breast-w	73.53	73.53	65.52	73.28	88.48	94.71	95.00
credit-a	73.33	74.93	55.51	73.66	80.72	85.62	84.53
diagnosis	69.38	64.38	50.00	65.63	81.88	94.79	95.21
haberman	70.59	57.50	73.53	66.04	69.59	73.20	73.53
heart-c	77.23	78.47	53.87	75.75	79.28	82.58	83.58
heart-h	71.88	64.14	63.94	65.25	77.82	80.20	78.66
heart-statlog	73.98	73.06	55.56	72.96	77.50	81.02	82.41
hepatitis	72.90	45.81	79.35	54.84	74.19	75.48	77.42
house-vote	74.02	52.41	61.38	65.29	82.99	88.05	89.20
income	73.21	73.50	51.25	73.21	74.33	76.29	75.88
ionosphere	72.06	53.87	64.13	66.07	74.91	80.61	76.95
labor	71.47	43.78	58.04	58.04	76.36	81.82	81.82
pima	79.63	79.98	65.11	80.18	80.73	73.21	71.45
splice	83.07	83.07	50.26	83.07	92.45	94.40	95.18
vote	75.46	75.34	61.38	74.20	87.99	93.62	89.20
z-alizadeh-sani	73.91	68.83	71.29	68.29	78.45	77.39	78.71
Average	73.73	66.83	62.59	69.81	78.83	81.69	81.66
Average ranking	4.3056	5.4444	5.5	5.1667	3	2.5833	2

model quality when the number of labelers is small. We summarize these experimental results as follows.

(1) When the number of labelers is small, Tables 1 and 5 show that our algorithms can improve the label quality no matter in the case of $J = 3$ or $J = 5$. When $J = 3$, the average label qualities obtained by the seven algorithms are 67.39% (MV), 56.02% (KOS), 62.25% (DS), 65.56% (RY), 73.68%

Table 6 The classification accuracy (%) comparisons with $J = 5$

Data set	MV	KOS	DS	RY	MNLDP	RSVMI^{w}	RSVMI^{e}
blood	76.2	76.2	76.2	76.2	76.2	76.2	76.74
breast-cancer	65.03	59.52	70.24	62.3	68.51	67.85	69.56
breast-w	94.99	95.42	65.51	94.42	96.29	76.68	78.25
credit-a	84.64	84.64	55.51	84.78	83.77	84.93	85.8
diagnosis	89.17	74.17	45.83	78.33	96.67	95	97.5
haberman	72.56	72.56	73.55	72.56	72.24	73.55	73.55
heart-c	77.16	69.52	49.27	71.84	77.25	81.51	82.51
heart-h	71.8	52.11	63.94	65.62	69.16	73.74	74.76
heart-statlog	76.67	77.78	55.56	77.78	77.41	83.7	82.59
hepatitis	78.06	71.61	79.35	81.29	79.35	81.29	81.29
house-vote	87.59	72.41	61.38	88.05	88.74	91.49	88.97
income	72.33	73.5	48	70.17	69.33	75.17	76.17
ionosphere	72.15	52.92	58.19	52.62	73.43	72.44	74.22
labor	68.48	58.79	58.85	55.69	68.2	75.59	75.59
pima	71.23	72.26	65.1	67.95	68.34	71.62	72.26
splice	82.92	82.92	49.8	82.92	86.83	93.74	95.83
vote	94.25	93.56	61.38	90.11	92.64	94.71	90.11
z-alizadeh-sani	74.09	74.34	71.29	73.02	72.44	77.96	78.12
Average	78.3	73.01	61.61	74.76	78.71	80.4	80.77
Average ranking	4.1944	4.75	5.8056	4.8333	4.0278	2.5833	1.8056

Table 7 The label quality (%) post-hoc comparisons with $J = 5$ [a)]

i	Algorithm	$z = (R_0 - R_i)/SE$	p
1	DS vs. RSVMI^{e}	4.860556	0.000001
2	KOS vs. RSVMI^{e}	4.744828	0.000002
3	RY vs. RSVMI^{e}	4.397645	0.000011
4	DS vs. RSVMI^{w}	4.050463	0.000051
5	KOS vs. RSVMI^{w}	3.934735	0.000083
6	RY vs. RSVMI^{w}	3.587553	0.000334
7	DS vs. MNLDP	3.471825	0.000517
8	KOS vs. MNLDP	3.356098	0.000791
9	MV vs. RSVMI^{e}	3.24037	0.001194
10	RY vs. MNLDP	3.008915	0.002622
11	MV vs. RSVMI^{w}	2.430278	0.015087
12	MV vs. MNLDP	1.85164	0.064078
13	MV vs. DS	1.620185	0.105193
14	MV vs. KOS	1.504458	0.132464
15	MNLDP vs. RSVMI^{e}	1.38873	0.164915
16	MV vs. RY	1.157275	0.24716
17	RSVMI^{w} vs. RSVMI^{e}	0.810093	0.417887
18	MNLDP vs. RSVMI^{w}	0.578638	0.562834
19	DS vs. RY	0.46291	0.643429
20	KOS vs. RY	0.347183	0.728454
21	KOS vs. DS	0.115728	0.907869

a) Nemenyi's procedure rejects those hypotheses that have an unadjusted p-value $\leqslant 0.002381$:

- DS vs. RSVMI^{e}
- KOS vs. RSVMI^{e}
- RY vs. RSVMI^{e}
- DS vs. RSVMI^{w}
- KOS vs. RSVMI^{w}
- RY vs. RSVMI^{w}
- MV vs. RSVMI^{e}

(MNLDP), 78.84% (RSVMI^{w}) and 81.33% (RSVMI^{e}). When $J = 5$, the average label qualities obtained by the seven algorithms are 73.73% (MV), 66.83% (KOS), 62.59% (DS), 69.81% (RY), 78.83% (MNLDP), 81.69% (RSVMI^{w}) and 81.66% (RSVMI^{e}). It can be seen from the average label qualities that when the number of labelers is reduced from 5 to 3, RSVMI^{e} and RSVMI^{w} can still maintain relatively moderate

Table 8 The classification accuracy (%) post-hoc comparisons with $J = 5$ [a)]

i	Algorithm	$z = (R_0 - R_i)/SE$	p
1	DS vs. RSVMI^{e}	5.554921	0
2	DS vs. RSVMI^{w}	4.474797	0.000008
3	RY vs. RSVMI^{e}	4.204766	0.000026
4	KOS vs. RSVMI^{e}	4.089039	0.000043
5	MV vs. RSVMI^{e}	3.317522	0.000908
6	RY vs. RSVMI^{w}	3.124643	0.00178
7	MNLDP vs. RSVMI^{e}	3.086067	0.002028
8	KOS vs. RSVMI^{w}	3.008915	0.002622
9	DS vs. MNLDP	2.468854	0.013555
10	MV vs. DS	2.237399	0.02526
11	MV vs. RSVMI^{w}	2.237399	0.02526
12	MNLDP vs. RSVMI^{w}	2.005944	0.044862
13	KOS vs. DS	1.465882	0.14268
14	DS vs. RY	1.350154	0.176966
15	RY vs. MNLDP	1.118699	0.263268
16	RSVMI^{w} vs. RSVMI^{e}	1.080123	0.280087
17	KOS vs. MNLDP	1.002972	0.315874
18	MV vs. RY	0.887244	0.374947
19	MV vs. KOS	0.771517	0.440401
20	MV vs. MNLDP	0.231455	0.816961
21	KOS vs. RY	0.115728	0.907869

a) Nemenyi's procedure rejects those hypotheses that have an unadjusted p-value $\leqslant 0.002381$:

- DS vs. RSVMI^{e}
- DS vs. RSVMI^{w}
- RY vs. RSVMI^{e}
- KOS vs. RSVMI^{e}
- MV vs. RSVMI^{e}
- RY vs. RSVMI^{w}
- MNLDP vs. RSVMI^{e}

performance.

(2) Tables 3 and 7 show the label quality post-hoc comparisons when the numbers of labelers are 3 and 5 respectively. The label quality post-hoc comparisons further demonstrate the superiority of our algorithm. From Table 3, we can see that RSVMI^{e} performs significantly better than MV, KOS, DS, RY and MNLDP. RSVMI^{w} also performs significantly better than MV, KOS, DS and RY. From Table 7, we can still reach similar conclusion.

(3) In addition to label quality, our method RSVMI still performs well in model quality. Tables 2 and 6 show the detailed classification accuracy results when the numbers of labelers are 3 and 5 respectively. When $J = 3$, the average model qualities of applying MV, KOS, DS, RY, MNLDP, RSVMI^{w} and RSVMI^{e} are 76.20%, 64.32%, 61.79%, 70.13%, 76.39%, 78.48% and 81.23%. When $J = 5$, the average model qualities of applying MV, KOS, DS, RY, MNLDP, RSVMI^{w} and RSVMI^{e} are 78.3%, 73.01%, 61.61%, 74.76%, 78.71%, 80.4% and 80.77%.

(4) Tables 4 and 8 respectively show the results of the model quality post-hoc comparisons under different experimental settings. Table 4 shows that when the number of labelers is 3, RSVMI^{e} performs significantly better than MV, KOS, DS, RY and MNLDP. RSVMI^{w} performs significantly better than KOS, DS, RY. Table 8 shows that when the number of labelers is 5, RSVMI^{e} performs significantly better than MV, KOS, DS, RY and MNLDP. RSVMI^{w} performs significantly better than DS, RY.

(5) In summary, according to the experimental results under different settings and the corresponding post-hoc Nemenyi tests, we can conclude that our algorithm RSVMI performs better than its competitors. Especially when the number of crowdsourced labels collected is relatively small, our algorithm is prominent. What is more, RSVMI^{e} is slightly better than RSVMI^{w}. There may be two reasons for this. First, there is a deviation in the estimation of the quality of each worker by using MV. Especially, the lower the number of workers, the worse the performance of MV, which makes $\overline{p}_i$ overestimated. Overestimated $\overline{p}_i$ results in the inaccurate estimate of q_i, which weakens the performance of the robust SVM. The second is that in the benchmark data experiments, the number of labels collected for each sample is the same so that $q_i, i = 1, 2, \ldots, n$ are the same in RSVMI^{w}. In fact, as mentioned above, different multiple noisy label sets reveal different information. Therefore, these two reasons affect the performance

Table 9 Description of five real-world crowdsourced data sets

Data set	Task	Instance	Positive	Negative	Labeler	Label	Feature
Leaves-t	tilia/oak	141	45	96	70	883	64
Leaves-e	eucalyptus/oak	140	45	95	66	930	64
Reuters-0	0/1	1420	934	486	37	3394	50
Income94	<50/>50	600	300	300	73	11999	14
Income94L10	<50/>50	600	300	300	67	6000	10

of RSVMI$^{\mathrm{w}}$ to a certain extent.

4.2 Experiments on real-world data sets

In this subsection, we ran our experiments on different real-world data sets including Leaves, Reuters, Income94, which were collected from Amazon Mechanical Turk (AMT) and are publicly available. In order to further verify the impact of the number of crowdsourced labels on the inference algorithms, we consider two strategies. The first is to directly use the original real-world data sets. The second strategy is that if a sample has more than three crowdsourced labels, we will randomly delete some labels so that each sample has at most three available crowdsourced labels.

The data sets Leaves and Income94 were downloaded from CEKA [36]. The task of the data set Leaves is to determine six kinds of leaves pictures. There are 384 samples described by 64 features in the Leaves data set. The data set Income94 comes from a traditional classification problem that uses 14 features to determine whether a person makes over \$50k/year. The data set provider extracted 300 positive samples and 300 negative samples from the original problem, and generated two new data sets. The first data set contains 600 samples with 14 features, and the second is to delete 4 features, so that each sample has only 10 features. Then these two data sets were submitted to the crowdsourcing platform to collect crowdsourced labels. In order to distinguish the two data sets, we call the former Income94 and the latter Income94L10. The data set Reuters was downloaded from the website of the paper [37]. The original data set Reuters-21578 [38] is a collection of manually categorized newswire stories with labels such as Acquisitions, Crudeoil, Earnings or Grain. The data set provider only considered the documents belonging to the ModApte split, and attached a constraint that the documents should not have multiple labels. This resulted in a total of 7016 documents with 8884 features distributed among 8 classes. Then 1799 documents were submitted to AMT for multiple labelers to label. Since the number of features is much larger than the number of samples, we use correlation-based feature selection (CFS) [39] to reduce the number of features to 50.

Because the original data sets Leaves and Reuters are multi-class data sets, in order to adapt the binary classification in this paper, we extract some binary data sets from original data sets. Two binary data sets from Leaves are denoted as Leaves-t and Leaves-e respectively. It should be noted that the Reuters data set is an unbalanced data set, and the total number of samples of class 0 and class 1 is 1501. Thus we only extract one binary data set from Reuters. The binary data set from Reuters is denoted as Reuters-0. Taking Leaves-t as an example, in the process of extraction, we first extract samples with real labels of tilia or oak. For these samples, we delete the crowdsourced labels that are not relevant to this classification task. That is to say, if a sample with the real label of oak is labeled as eucalyptus by a worker, this crowdsourced label will not appear in Leaves-t. It should be noted that if the crowdsourced labels of a sample are always irrelevant to the current classification task, the sample will not appear in the final binary data set. The detailed data characteristics of these five data sets are listed in Table 9. Take one of these five data sets as an example, the task of "Leaves-t" is to distinguish leaves of tilia and oak, which contains 45 positive samples (tilia) and 96 negative samples (oak). In this data set, 70 workers provided 883 labels.

Figure 3 shows the detailed comparison results of seven algorithms on five real-world crowdsourced data sets. Figures 3(a) and (b) show the label qualities and model qualities based on seven inference methods on five original data sets. Figures 3(c) and (d) show the label qualities and model qualities based on seven inference methods under the second strategy (randomly deleting some labels so that each sample has at most three available crowdsourced labels). From Figure 3, we can see that our methods perform overall better than the other five algorithms. Although the second strategy has fewer crowdsourced labels than the first strategy, especially for the Income94 and Income94L10 data sets, our algorithm can still maintain a good result. This also proves that RSVMI can achieve our goal: to maintain moderate performance

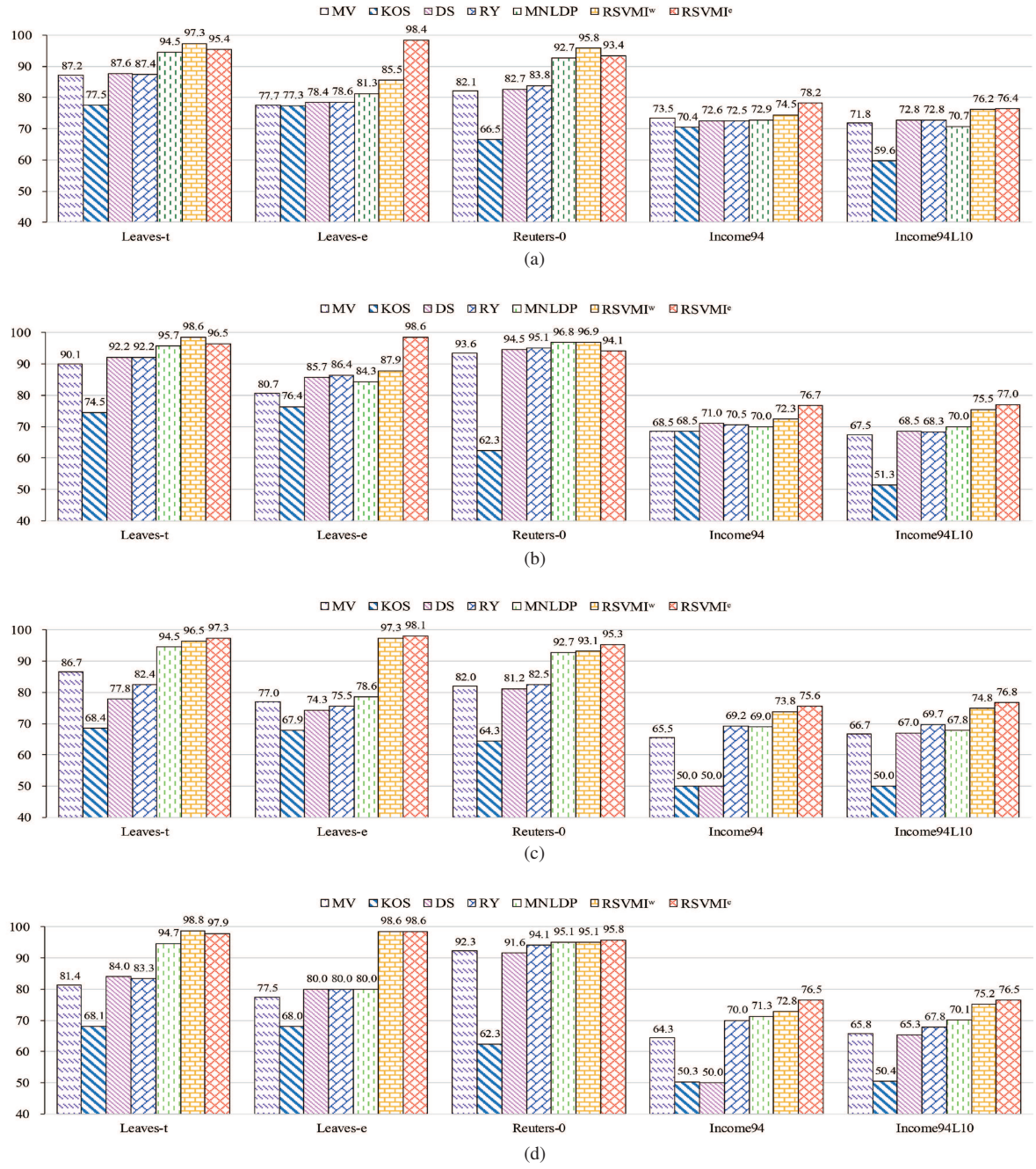

Figure 3 (Color online) The detailed comparison results on three real-world crowdsourced data sets. (a) Label quality comparisons on original real-world data sets; (b) classification accuracy comparisons on original real-world data sets; (c) label quality comparisons on modified real-world data sets; (d) classification accuracy comparisons on modified real-world data sets.

while reducing labeling costs.

5 Conclusion and future work

In order to reduce labeling costs and maintain moderate performance simultaneously, this paper proposes a novel ground truth inference algorithm RSVMI for crowdsourcing learning based on the label noise robust SVM. By modifying the optimization problem, the robust SVM can use the label noise to establish the decision hyperplane. It is worth noting that despite the modification of the optimization goal, we still only need to solve a convex optimization problem. In order to apply the robust SVM to crowdsourced

data, a crucial issue is how to estimate the noise level of integrated labels. This paper proposes two methods to estimate the noise level of integrated labels based on a binomial distribution and a modified sigmoid function respectively. The experimental results show that RSVMI can achieve better performance than its competitors when the number of labelers is small.

As shown in Section 3, RSVMI must estimate the probability of each initial integrated label being flipped. Although two estimation methods are proposed in this paper, they are still somewhat rough. We believe that sophisticated techniques of estimation can improve the performance of RSVMI, and this will be one of our future work. In addition, although SVM can use kernel trick to deal with nonlinear separable problems, there are still many limitations. For example, it is difficult for us to select the appropriate kernel when we do not know the internal structure of the data in advance. One potential future direction is that designing a more complicated but still trainable label noise robust classifier for crowdsourcing learning.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. U1711267) and Fundamental Research Funds for the Central Universities (Grant No. CUGGC03).

References

1 Layale C, Hazem H. A survey of ground-truth in emotion data annotation. In: Proceedings of the 10th Annual IEEE International Conference on Pervasive Computing and Communications, Lugano, 2012. 697–702
2 Liu B. Sentiment analysis and opinion mining. Synthesis Lectures Human Language Technol, 2012, 5: 1–167
3 Sheng V S, Provost F J, Ipeirotis P G. Get another label? Improving data quality and data mining using multiple, noisy labelers. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, 2008. 614–622
4 Sheng V S. Simple multiple noisy label utilization strategies. In: Proceedings of the 11th IEEE International Conference on Data Mining, Vancouver, 2011. 635–644
5 Li H, Yu B. Error rate bounds and iterative weighted majority voting for crowdsourcing. 2014. ArXiv: 1411.4086
6 Tian T, Zhu J. Max-margin majority voting for learning from crowds. In: Proceedings of Annual Conference on Neural Information Processing Systems, Montreal, 2015. 1621–1629
7 Zhang H, Jiang L, Xu W. Differential evolution-based weighted majority voting for crowdsourcing. In: Proceedings of the 15th Pacific Rim International Conference on Artificial Intelligence, Nanjing, 2018. 228–236
8 Jiang L, Zhang H, Tao F, et al. Learning from crowds with multiple noisy label distribution propagation. IEEE Trans Neural Networks Learn Syst, 2022, 33: 6558–6568
9 Tao F, Jiang L, Li C. Label similarity-based weighted soft majority voting and pairing for crowdsourcing. Knowl Inf Syst, 2020, 62: 2521–2538
10 Dawid A P, Skene A M. Maximum likelihood estimation of observer error-rates using the EM algorithm. Appl Stat, 1979, 28: 20–28
11 Raykar V C, Yu S, Zhao L H, et al. Learning from crowds. J Mach Learn Res, 2010, 11: 1297–1322
12 Demartini G, Difallah D E, Cudré-Mauroux P. Zencrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In: Proceedings of the 21st World Wide Web Conference, Lyon, 2012. 469–478
13 Zhang J, Wu X, Sheng V S. Imbalanced multiple noisy labeling. IEEE Trans Knowl Data Eng, 2015, 27: 489–503
14 Wu M, Li Q, Zhang J, et al. A robust inference algorithm for crowd sourced categorization. In: Proceedings of the 12th International Conference on Intelligent Systems and Knowledge Engineering, Nanjing, 2017. 1–6
15 Karger D R, Oh S, Shah D. Iterative learning for reliable crowdsourcing systems. In: Proceedings of the 25th Annual Conference on Neural Information Processing Systems, Granada, 2011. 1953–1961
16 Zhang J, Sheng V S, Wu J, et al. Multi-class ground truth inference in crowdsourcing with clustering. IEEE Trans Knowl Data Eng, 2016, 28: 1080–1085
17 Ruiz P, Morales-Álvarez P, Molina R, et al. Learning from crowds with variational Gaussian processes. Pattern Recogn, 2019, 88: 298–311
18 Li C, Sheng V S, Jiang L, et al. Noise filtering to improve data and model quality for crowdsourcing. Knowledge-Based Syst, 2016, 107: 96–103
19 Whitehill J, Ruvolo P, Wu T, et al. Whose vote should count more: optimal integration of labels from labelers of unknown expertise. In: Proceedings of the 23rd Annual Conference on Neural Information Processing Systems, Vancouver, 2009. 2035–2043
20 Rodrigues F, Pereira F C, Ribeiro B. Gaussian process classification and active learning with multiple annotators. In: Proceedings of the 31st International Conference on Machine Learning, Beijing, 2014. 433–441
21 Zhang M L, Yu F, Tang C Z. Disambiguation-free partial label learning. IEEE Trans Knowl Data Eng, 2017, 29: 2155–2167
22 Rodrigues F, Pereira F C. Deep learning from crowds. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, the 30th Innovative Applications of Artificial Intelligence, and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, 2018. 1611–1618
23 Guan M Y, Gulshan V, Dai A M, et al. Who said what: modeling individual labelers improves classification. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, the 30th innovative Applications of Artificial Intelligence, and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, 2018. 3109–3118
24 Dalvi N N, Domingos P M, Sumit M, et al. Adversarial classification. In: Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, 2004. 99–108
25 Biggio B, Nelson B, Laskov P. Support vector machines under adversarial label noise. In: Proceedings of the 3rd Asian Conference on Machine Learning, Taiwan, 2011. 97–112
26 Cortes C, Vapnik V. Support-vector networks. Mach Learn, 1995, 20: 273–297
27 Ye Z K, Li L Z, Situ H Z, et al. Quantum speedup of twin support vector machines. Sci China Inf Sci, 2020, 63: 189501
28 Zhang J, Sheng V S, Nicholson B, et al. CEKA: a tool for mining the wisdom of crowds. J Mach Learn Res, 2015, 16: 2853–2858

29 Witten I H, Frank E, Hall M A. Data Mining: Practical Machine Learning Tools and Techniques. 3rd ed. San Francisco: Morgan Kaufmann, 2011
30 Dheeru D, Casey G. UCI machine learning repository. 2017. http://archive.ics.uci.edu/ml
31 Jiang L, Zhang L, Yu L, et al. Class-specific attribute weighted naive Bayes. Pattern Recogn, 2019, 88: 321–330
32 Jiang L, Zhang L, Li C, et al. A correlation-based feature weighting filter for naive Bayes. IEEE Trans Knowl Data Eng, 2019, 31: 201–213
33 Alcalá-Fdez J, Fernández A, Luengo J, et al. KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. J Multiple-Valued Logic Soft Comput, 2011, 17: 255–287
34 Demsar J. Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res, 2006, 7: 1–30
35 Garcia S, Herrera F. An extension on statistical comparisons of classifiers over multiple data sets for all pairwise comparisons. J Mach Learn Res, 2008, 9: 2677–2694
36 Zhang J, Wu X, Sheng V S. Learning from crowdsourced labeled data: a survey. Artif Intell Rev, 2016, 46: 543–576
37 Rodrigues F, Lourenco M, Ribeiro B, et al. Learning supervised topic models for classification and regression from crowds. IEEE Trans Pattern Anal Mach Intell, 2017, 39: 2409–2422
38 Lewis D D. Evaluating text categorization. In: Proceedings of the Workshop on Speech and Natural Language, 1991. 312–318
39 Hall M A. Correlation-based feature selection for discrete and numeric class machine learning. In: Proceedings of the 17th International Conference on Machine Learning, 2000. 359–366