Learning With Label Proportions via NPSVM

Zhiquan Qi, Bo Wang, Fan Meng, and Lingfeng Niu

***Abstract*—Recently, learning from label proportions (LLPs), which seeks generalized instance-level predictors merely based on bag-level label proportions, has attracted widespread interest. However, due to its weak label scenario, LLP usually falls into a transductive learning framework accounting for an intractable combinatorial optimization issue. In this paper, we propose a brand new algorithm, called LLPs via nonparallel support vector machine (LLP-NPSVM), to facilitate this dilemma. To harness satisfactory data adaption, instead of transductive learning fashion, our scheme determined instance labels according to two nonparallel hyper-planes under the supervision of label proportion information. In a geometrical view, our approach can be interpreted as an alternative competitive method benefiting from large margin clustering. In practice, LLP-NPSVM can be efficiently addressed by applying two fast sequential minimal optimization paths iteratively. To rationally support the effectiveness of our method, finite termination and monotonic decrease of the proposed LLP-NPSVM procedure were essentially analyzed. Various experiments demonstrated our algorithm enjoys rapid convergence and robust numerical stability, along with best accuracies among several recently developed methods in most cases.**

***Index Terms*—*k*-plane clustering, learning with label proportions (LLPs), nonparallel support vector machine (NPSVM).**

I. Introduction

SUPERVISED, semi-supervised, and unsupervised learnings are three mainly topics and fashions in the machine learning society [1]–[11]. Nevertheless, it is arguable that many real life problems fail in being simply abstracted into these three machine learning communities, especially when the labels cannot be fully accessed and none of them can be explicitly determined. Take an investment strategy exploration in selling certain sort of product as an example.

A marketing company plans to increase its profit in sales through sending out discount coupons. Normally, it is helpful to improve profit when and only when sending coupons to customers who would just buy the discount products. However, we cannot identify this sort of potential clients explicitly. Instead, only proportions of these people in some certain groups are known somehow. Let us say that about 75% housewives and only 15% businessmen will be the objective customers in common sense. What can we do with this information to promote sales more precisely?

Not only in the social science, similar problems are also ubiquitous in natural science world. Consider the following often occurred issue in health-care research.

In view of the privacy protection, only the proportion of diagnosed diseases in each ZIP code area is literally available to the public. Suppose that you were an epidemic expert and hoped to reliably diagnose for every single latent patient. How can you successfully learn these individual labels (disease or nondisease) from regular test results and group-level label proportions?

Besides, in spam filtering, democratic election, similar interesting puzzles are frequently encountered, which can be informally concluded into how to obtain a reliable classifier merely using label proportions information and instances' features. In other words, this issue can be formally regarded as learning from label proportions (LLPs) problem, which has generated considerable recent research interests. Generally speaking, unlike the machine learning problems mentioned above, in LLP problem, training instances are provided in manner of bags, and only the proportion of each class in every individual bag is available. On the other hand, the learning task is to predict labels of new individual instances. Explicitly, LLP problem can be described in the following specific fashion.

Consider a binary classification problem. Suppose that the training set $\{\boldsymbol{x_i}\}_{i=1}^{N} \subset \mathcal{X}$ is given in the form of K disjoint bags, that is

$$\{\boldsymbol{x_i}|i \in \mathcal{B}_k\}_{k=1}^{K}, \bigcup_{k=1}^{K} \mathcal{B}_k = \{1, 2, \ldots, N\}, \mathcal{B}_k \cap \mathcal{B}_l = \emptyset, \ \forall k \neq l.$$

Different from unsupervised learning situation, label information is abstracted in bag level somehow. Particularly, the proportion of positive class points in every bag is available. In light of this, we explicitly denote $y_i^* \in \{-1, +1\}, i = 1, 2, \ldots, N$ the unknown ground truth label for every instance and measure proportion information in the kth bag by $p_k := (|\{i|i \in \mathcal{B}_k, y_i^* = 1\}|/|\mathcal{B}_k|) \in [0, 1], \forall k$, correspondingly. Our eventual goal is to learn a classifier based on $\{\boldsymbol{x_i}\}_{i=1}^{N}$ and $\{p_k\}_{k=1}^{K}$ in order to predict label in instance level.

Manuscript received February 19, 2016; revised June 8, 2016; accepted July 25, 2016. Date of publication August 24, 2016; date of current version September 14, 2017. This work was supported in part by the National Natural Science Foundation of China under Grant 61402429, Grant 61472390, Grant 11271361, and Grant 11331012, in part by the Key Project of National Natural Science Foundation of China under Grant 71331005, and in part by the Major International (Regional) Joint Research under Project 71110107026. This paper was recommended by Associate Editor F. Karray. *(Corresponding author: Lingfeng Niu.)*

Z. Qi, F. Meng, and L. Niu are with the Research Center on Fictitious Economy and Data Science, Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing 100190, China (e-mail: niulf@ucas.ac.cn).

B. Wang is with the Research Center on Fictitious Economy and Data Science, Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Information Technology and Management, University of International Business and Economics, Beijing 100029, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2016.2598749

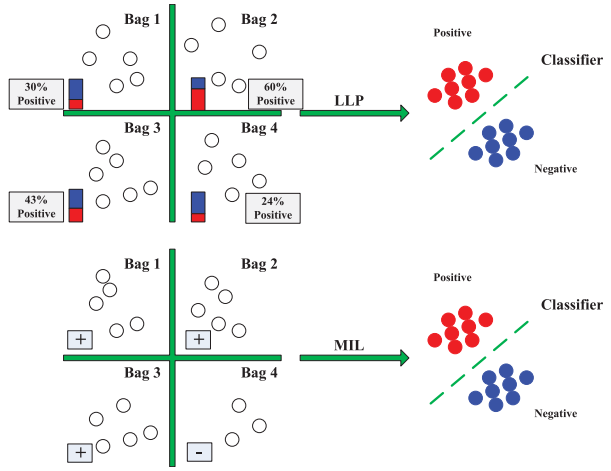


Fig. 1. Comparison between LLP and MIL problems.

As a special learning with bags task for binary classification, LLP can be associated with another slight different learning problem: multi-instance learning (MIL) [12]–[19], in which instance-level labels are not available either. Instead of proportion information, in MIL problem, only two sorts of bags, i.e., the positive bag (containing at least one positive instance) and negative bag (containing no positive instance), are involved. From this point of view, MIL can be treated as a degeneration of LLP with extremely high or low proportions. Nonetheless, in general LLP problem setting, one cannot tell explicit label of any single instance. By contrast, this is somewhat inconsistent to MIL, where one can label all the instances in negative bags without hesitation. Thus, MIL can be transferred into a standard semi-supervised learning problem and be solved by constrained concave–convex procedure [20], [21] bearing its transductive learning framework intrinsically. Fig. 1 sketches these two similar learning problems.

A. Related Work

While the natural born transductive scenario is knotty, as an alternative learning task, LLP has indeed outlined a promising middle ground between supervised and unsupervised learning. The related researches mainly fell into four regimes. The first one could be traced back to the work of Kuck and de Freitas [22]. They provided an Markov Chain Monte Carlo algorithm to handle this problem that accounted for uncertainty in model parameters and unknown individual labels as well. Whereas, its efficiency was severely limited by the complexity of the essential weak label information. Second, learning from aggregate views was introduced by Chen *et al.* [23] together with different learning methods for a special case called learning from projections and counts. Meanwhile, conditional class estimations were restricted to match observed ones. Similar work was also mentioned in [24].

Nevertheless, Quadrianto *et al.* [25] applied consistent estimators which could reconstruct the correct labels with high probability in an uniform convergence sense. In detail, they assumed the distribution of labels conditioned on the features via a conditional exponential model: $p(y|\boldsymbol{x},\theta) = \exp(\langle\phi(\boldsymbol{x},y),\theta\rangle - g(\theta|\boldsymbol{x}))$. Here, $g(\theta|\boldsymbol{x}) = \log\sum_{\boldsymbol{y}\in\mathcal{Y}}\exp\langle\phi(\boldsymbol{x},\boldsymbol{y}),\theta\rangle$ was called the log-partition function, and $\phi(\boldsymbol{x},\boldsymbol{y})$ denoted a feature map from $\mathcal{X}\times\mathcal{Y}$ to a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$. The parameter θ needed to be estimated based on labels proportion information. However, a key assumption, $p(\boldsymbol{x}|\boldsymbol{y},i) = p(\boldsymbol{x}|\boldsymbol{y})$, where i denoted the ith bag, seemed to be far too unrealistic. In addition, the estimation process strongly depended on empirical means and expectations scale.

Later, Rüping [26] proposed an parametric LLP solver outperformed previous approaches by combining support vector regression and inverse classifier calibration. In fact, mean of each bag was treated as a "super-instance" and produced a soft label through corresponding label proportion.

Unfortunately, this hasty process by integrating property in bag level was too general to result in acceptable performance in some uncommon cases argued by Yu *et al.* [27].

Besides, Stolpe and Morik [28] developed a clustering-based approach to Tame LLP puzzle. However, it suffered from the extremely high computing complexity. In addition, Fan *et al.* [29] presented theoretical analysis to associate LLP with supervised learning, through giving a sufficient condition for learnable binary classification scenario. What is more, they conducted a fruitful framework on how to build generative classifiers by density estimation. The experimental results on benchmark data sets succeeded in promising performance. On the other hand, Patrini *et al.* [30] offered a fast learning algorithm to estimate the mean operator via a manifold regularizer with guaranteed approximation bounds.

Recently, an effective model based on support vector machine (SVM), called $\propto$SVM [27], [31], overwhelmingly outperformed these former known methods in most situations with carefully trick setting. A maximum margin framework was employed to optimize over the unknown instance labels and the known label proportions simultaneously. Through alleviating restrictive assumptions on the data, either parametric or generative (many assumptions cannot hold for general real-world applications), $\propto$SVM discovered a more universal framework for LLP, which iteratively polished the solution with annealing loop. In detail, this LLP algorithm based on the maximum margin framework can be expressed as

$$\min_{\boldsymbol{y}\in\mathcal{C},\boldsymbol{w},b}\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}L\Big(y_i,\boldsymbol{w}^\top\varphi(\boldsymbol{x_i})+b\Big) + C_p\sum_{i=1}^{l}L_p(\tilde{p}_i(\boldsymbol{y}),p_i) \quad (1)$$

where $\mathcal{C}$ indicated $\{-1,+1\}^l$, $L(\cdot)\geq 0$ was the loss function inherited from supervised learning and $L_p(\cdot)\geq 0$ was a 1-Lipshitz loss function to penalize the error between the prior ground truth label proportions and the posteriori ones.

Although $\propto$SVM can implement optimization in searching $\boldsymbol{y}$, $\boldsymbol{w}$, and b by various methods, it is spiked in some complicated applications for its predestined disadvantages. Particularly, weak label information indubitably results in a nonconvex integer programming for this problem, which

is often NP-hard. That is to say, the optimization problem in (1) is substantially combinatorial. As a result, this obstacle impedes us from obtaining adequately approximate result in limited time.

In addition, other references can be found in [28] and [32]–[37]. Currently, LLP has also been enormously applied in marketing, election, span filtering [22], visual attribute modeling [38], [39], video event detection [40], predicting income based on census data [31], and so on.

B. Motivation

As is introduced above, the essential difference among the former methods is that different tactics they used to tradeoff between the deviations in the instance-level feature information and bag-level label proportions information. Although none of them can dominate in all circumstances, there is no doubt that most of these successes can be attributed to the exploration of between-class information. For example, a maximum margin framework explicitly modeling the latent unknown instance labels together with the known group label proportions (called $\propto$SVM) [27], can greatly boost accuracy in predicting instance level. On one hand, this remarkable advantage lies in the introduction of large margin mechanism, which fully implements the data's between-class information. On the other hand, taking account of proportion information, empirical proportion risk minimization principle (EPRMP) was proposed to suppress and alleviate the inconsistency with prior proportion information. Relying on a systemic compromise between large margin principle (LMP) and EPRMP, $\propto$SVM achieved a considerable competitiveness in performance.

It is true that between-class information plays an important role in LLP problem under separation assumption, but sufficient illustrations argue that data distribution information itself is seemed to be more indispensable in obtaining a generalized classifier, which will be exactly achieved by our method proposed in this paper. In stark contrast to this observation, the neglect of distribution tendency is prone to commit a fallacy of unseen data and will precipitate a less generalized classifier. Furthermore, implanting nonparallel hyperplane clustering method [11], [41] into binary classification problem happens to profoundly integrate between-class separation and data distribution information to hybridize promising classifiers.

To illustrate our method, Fig. 2 reports an intuitive explanation of LMP and nonparallel hyperplane clustering. Here, we assume the data is generated according to two 2-variate Gaussian distributions in $\mathbb{R}^2$ with parameters setting as $\mu_1 = (0, 4)$, $\sigma_1 = \begin{bmatrix} 5 & -4 \\ -4 & 5 \end{bmatrix}$ and $\mu_2 = (20, 4)$, $\sigma_2 = \begin{bmatrix} 5 & 4 \\ 4 & 5 \end{bmatrix}$, respectively. The blue line denotes the result of $\propto$SVM, in the meantime, the red and cyan lines denote the result of our method. Consequently, when a new instance comes, $\propto$SVM offers its label according to one single blue line, yet our method tries to obtain its label based on the nearest distance principle to two nonparallel lines. By contrast, two hyper-planes are obtained by minimizing the squared sum of Euclidean distances from each instance to the nearest hyperplane. That is to say, traditional point-type cluster centers are replaced by hyper-planes, which excellently describes the ground truth data distribution. In this way, our method yields the posteriori label of a new instance relying on two nonparallel lines. From Fig. 2, we can tell that these two nonparallel lines are more likely to seize and represent appropriate distribution of data as is described above. To get a deeper understanding of this character, it can be easily learned that when there are much more instances sampling from two Gaussian distributions, these two individual groups will overlap. Admittedly, between-class-based $\propto$SVM shows insufficient utilization of this observation. At the same time, two hyper-planes are incorporated into decision functions in two smaller problems, which is a directly implementation for LLP problem.

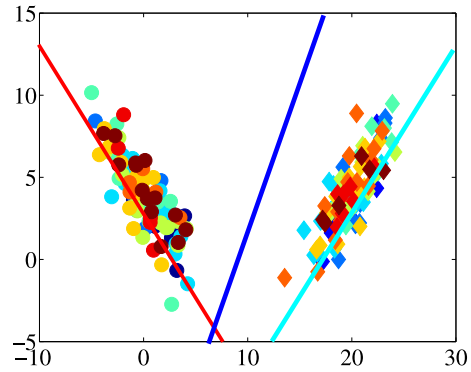


Fig. 2. Intuitive explanation of LMP and nonparallel hyper-planes idea. The $\circ$ denotes the positive class, which is generated by the normal distribution of parameters μ_1, σ_1, and the $\Diamond$ denotes the negative class, which is generated by the normal distribution of parameters: μ_2 and σ_2. The different colors denotes different bags. The blue line denotes the results of $\propto$SVM, the red and cyan lines denote the result of our method.

In practice, two smaller models first learn to construct two nonparallel hyper-planes to recover data distribution information. Then, labels in instance level can be supervised under bag proportion errors minimization principle. This procedure enables us to perform two sequential minimal optimization (SMO) paths iteratively and distributively, which has been repeatedly reported to drastically reduce the computation time. All experiments in Section IV indicate our algorithm is superior to current methods with rapid convergence and robust numerical stability.

The remaining parts of this paper are organized as follows. We first introduce the background in Section II, and then give our new algorithm: LLPs via nonparallel SVM (LLP-NPSVM) in Section III. All experiment results are shown in Section IV. Concluding remarks are summarized in Section V.

II. Background

In this section, we briefly introduce the nonparallel SVM (NPSVM) and k-plane clustering, both of which are derived by our method.

A. Nonparallel SVM

Consider binary classification problem with the training set

$$T = \{(\boldsymbol{x_i}, +1)\}_{i=1}^{p} \bigcup \{(\boldsymbol{x_j}, -1)\}_{j=p+1}^{p+q} \tag{2}$$

where $\boldsymbol{x_i} \in \mathbb{R}^n, i = 1, \ldots, p+q$. With the generalized twin SVM (TWSVM) [42] mechanism, incorporating sparsity through ε-band and structural risk minimization principle, NPSVM receives several remarkable advantages compared with the existing TWSVMs [43]. Particularly, for linear classification problem, NPSVM seeks two nonparallel hyper-planes

$$(\boldsymbol{w_+} \cdot \boldsymbol{x}) + b_+ = 0 \text{ and } (\boldsymbol{w_-} \cdot \boldsymbol{x}) + b_- = 0 \tag{3}$$

by solving the two problems

$$\begin{aligned} \min_{\boldsymbol{w_+}, b_+, \boldsymbol{\eta_+^{(*)}}, \boldsymbol{\xi_-}} \quad & \frac{1}{2}\|\boldsymbol{w_+}\|^2 + c_1 \sum_{i=1}^{p} \left(\eta_i + \eta_i^*\right) + c_2 \sum_{j=p+1}^{p+q} \xi_j \\ \text{s.t.} \quad & -\varepsilon - \eta_i^* \le (\boldsymbol{w_+} \cdot \boldsymbol{x_i}) + b_+ \le \varepsilon + \eta_i, \ \forall i \\ & \left(\boldsymbol{w_+} \cdot \boldsymbol{x_j}\right) + b_+ \le -1 + \xi_j, \ \forall j \\ & \eta_i, \eta_i^* \ge 0, \ \forall i, \ \xi_j \ge 0, \ \forall j \end{aligned} \tag{4}$$

$$\begin{aligned} \min_{\boldsymbol{w_-}, b_-, \boldsymbol{\eta_-^{(*)}}, \boldsymbol{\xi_+}} \quad & \frac{1}{2}\|\boldsymbol{w_-}\|^2 + c_3 \sum_{i=p+1}^{p+q} \left(\eta_i + \eta_i^*\right) + c_4 \sum_{j=1}^{p} \xi_j \\ \text{s.t.} \quad & -\varepsilon - \eta_i^* \le (\boldsymbol{w_-} \cdot \boldsymbol{x_i}) + b_- \le \varepsilon + \eta_i, \ \forall i \\ & \left(\boldsymbol{w_-} \cdot \boldsymbol{x_j}\right) + b_- \ge 1 - \xi_j, \ \forall j \\ & \eta_i, \eta_i^* \ge 0, \ \forall i, \ \xi_j \ge 0, \ \forall j. \end{aligned} \tag{5}$$

Actually, the first constraint in (4) and (5) implements regression with respect to ε insensitive loss function. Meanwhile, the second constraint is related to the requirement far away from the alternative class. Instead of solving (4) and (5) directly, NPSVM applied SMO to their dual Quadratic Programming Problems with a minor modification.

B. k-Plane Clustering

By replacing point-type cluster centers with hyper-planes, Bradley and Mangasarian [41] explored this considerable clustering method. Clustering around planes appears to have advantages over clustering around points. Explicitly, consider a set $\mathcal{A}$ of m points in the n-dimensional real space $\mathbb{R}^n$ represented by the matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$. Group $\mathcal{A}$ into k clusters according to the following nonconvex minimization problem. Determine k cluster planes in $\mathbb{R}^n$

$$P_l := \left\{\boldsymbol{x} \in \mathbb{R}^n | \boldsymbol{x} \cdot \boldsymbol{w_l} = \gamma_l\right\}, l = 1, 2, \ldots, k \tag{6}$$

which minimize the sum of the squares of distances from each point in $\mathcal{A}$ to a nearest plane P_l. It alternates between assigning points to a nearest cluster plane (cluster assignment) and, for a given cluster, computing a cluster plane that minimizes the sum of the squares of distances to all points in the cluster (cluster update).

III. NPSVM FOR LLP

First, notations that simplify the statement need to be defined. Let $d^+(\boldsymbol{x}) = |\boldsymbol{w}_+^\top \boldsymbol{x} + b_+|, d^-(\boldsymbol{x}) = |\boldsymbol{w}_-^\top \boldsymbol{x} + b_-|$ be two measurements for refined distances between any arbitrary point $\boldsymbol{x}$ and two nonparallel hyper-planes in a certain iteration. Meanwhile, let $\{\{(\boldsymbol{x_i^j})\}_{i=1}^{n_j}, p_j\}_{j=1}^{m}$ represent the bags with label proportions and $\sum_{j=1}^{m} n_j = l$. Also, let $(1/l)\sum_{j=1}^{m} p_j n_j = p_+$.

A. Algorithm

First, we describe LLP-NPSVM as a standard framework and add annealing loop to avoid being attracted by local optima.

Specifically, d_k^+ and d_k^- are indicated to these two distance measures in iteration k. For simplicity, similar to the methodology in [41], we conclude the proposed algorithm in two steps, label assignment and label update. In every single bag j, according to the proportion preservation principle, i.e., EPRM, labels switch will occur when nearest assignment strategy fails in recovering labels in the last one. In other words, label update procedure conflicts to label assignment result in some iteration. For any point $\boldsymbol{x_j^i}$, in the kth iteration, $d_k^-(\boldsymbol{x_j^i}) - d_k^+(\boldsymbol{x_j^i})$ can be regarded as a key measure for label assignment, regardless of the label assignment result in the $(k-1)$th iteration. Particularly, assigned the labels of top-$n_j p_j$ instances with respect to $d_k^-(\boldsymbol{x_j^i}) - d_k^+(\boldsymbol{x_j^i})$ as $+1$, without loss of generality, we denote them as $\{\boldsymbol{x_i^j}\}_{i \in I(k)}$.

B. Finite Termination for Label Assignment

At first, we explicitly define the objective function in iteration k based on d_k^+ and d_k^- as follows:

$$\text{Obj}_k^a = \sum_{j=1}^{m} \left[\sum_{i \in I(k)} d_k^+\left(\boldsymbol{x_j^i}\right) + \sum_{i \in I(k)'} d_k^-\left(\boldsymbol{x_j^i}\right) \right]. \tag{7}$$

Obviously, proportion information and label assignment procedure are both in bag level, which allows us to consider the properties of objective function in each individual bag. First of all, we can prove the strict decrease of objective function with respect to label assignment.

Lemma 1: The label assignment procedure in Algorithm 1 renders a strictly decreasing path for the objective function.

Proof: Consider the proportion preservation principle as a strict constraint for this procedure. There are some pairwise label switches whenever the top-$n_j p_j$ labeling is inconsistent between two sequent iterations, for example between kth and $(k+1)$th iterations. For the simplicity's sake, let us suppose that there is only one pair of label switch, which means one positive point and one negative point in the kth iteration should be relabeled as the opposite classes in the $(k+1)$th iteration simultaneously.

Without loss of generality, assume that $d_k^-(\boldsymbol{x_j^s}) - d_k^+(\boldsymbol{x_j^s}) < d_k^-(\boldsymbol{x_j^t}) - d_k^+(\boldsymbol{x_j^t})$. Here, s and t are positive and negative label indexes in the kth iteration, respectively. As we can see, it follows that $d_k^-(\boldsymbol{x_j^s}) + d_k^+(\boldsymbol{x_j^t}) < d_k^+(\boldsymbol{x_j^s}) + d_k^-(\boldsymbol{x_j^t})$, which leads to a strict decrease of objection function value according to (7). ■

Algorithm 1 LLP-NPSVM

Require: Randomly initialize $y_i \in \{-1, 1\}, i = 1, 2, \cdots, l$, keeping $\frac{1}{l}\sum_{i=1}^{l}\frac{y_i+1}{2} = p_+$. $c_1 = 10^{-5}p_kC, c_2 = 10^{-5}(1-p_k)C$.
Apply linear NPSVM (4) and (5) to $\{\boldsymbol{x_i}, y_i\}_{i=1}^{l}$ to obtain two nonparallel hyper-planes $(\boldsymbol{w_+}^T, b_+)$ and $(\boldsymbol{w_-}^T, b_-)$;
while $(c_1 < C)\&\&(c_2 < C)$ **do**
  $c_1 = c_3 = min\{(1+\Delta)c_1, C\}$;
  $c_2 = c_4 = min\{(1+\Delta)c_2, C\}$;
  Let $dist_{current} = dist_{switch} = 0$ and $count = 0$;
  while $(dist_{current} \geq dist_{switch})\&\&(count < 2)$ **do**
    $count = count + 1$;
    repeat
      Relabel bags $\{\{(\boldsymbol{x_i^j})\}_{i=1}^{n_j}, p_j\}_{j=1}^{m}$ in instance level according to the following rule (**label assignment**):
      Sort $d^-(\boldsymbol{x_i^j}) - d^+(\boldsymbol{x_i^j})$;
      Label the Top-n_jp_j instances as $+1$;
      Label the rest instances as -1;
      Update $y_i, i = 1, 2, \cdots, l$;
      Apply linear NPSVM (4) and (5) to new labeled data $\{\boldsymbol{x_i}, y_i\}_{i=1}^{l}$ to obtain two nonparallel hyper-planes $(\boldsymbol{w_+}^T, b_+)$ and $(\boldsymbol{w_-}^T, b_-)$ (**label update**);
    until The change of objective function is smaller than some predefined threshold τ
      Compute $dist_{current}$;
      Switch labels and compute $dist_{switch}$ (**switch strategy**);
  end while
end while

Subsequently, we will show that this label assignment procedure is irreversible, which means the switch occurring in the kth iteration will not be withdrawn in the following iterations. Due to the finity of points in every bag, this leads to a finite termination. For simplicity, we choose equal weights for NPSVM in label update procedure. In addition, let the regularization item be $\|\boldsymbol{w}\|^2 = 1$, i.e., scaling $\boldsymbol{w}$. Moreover, let $I(k)$ be the indexes set for the positive points in iteration k, $I(k)'$ be the indexes set for negative ones in iteration k. Based on the optimality of $(\boldsymbol{w}, b)^\top$, let $0 < c_a, c_b < 1$, there are four equations should be introduced

$$\sum_{i\in I(k)} d_k^+\left(\boldsymbol{x_j^i}\right) - c_a\sum_{i\in I(k)'} d_k^+\left(\boldsymbol{x_j^i}\right) + \gamma_1 = \sum_{i\in I(k)} d_{k+1}^+\left(\boldsymbol{x_j^i}\right) - c_a\sum_{i\in I(k)'} d_{k+1}^+\left(\boldsymbol{x_j^i}\right) \tag{8}$$

$$\sum_{i\in I(k)'} d_k^-\left(\boldsymbol{x_j^i}\right) - c_b\sum_{i\in I(k)} d_k^-\left(\boldsymbol{x_j^i}\right) + \gamma_2 = \sum_{i\in I(k)'} d_{k+1}^-\left(\boldsymbol{x_j^i}\right) - c_b\sum_{i\in I(k)} d_{k+1}^-\left(\boldsymbol{x_j^i}\right) \tag{9}$$

$$\sum_{i\in I(k),i\neq s} d_{k+1}^+\left(\boldsymbol{x_j^i}\right) + d_{k+1}^+\left(\boldsymbol{x_j^t}\right) - c_a\sum_{i\in I(k)',i\neq t} d_{k+1}^+\left(\boldsymbol{x_j^i}\right) - c_a d_{k+1}^+\left(\boldsymbol{x_j^s}\right) + \gamma_3 = \sum_{i\in I(k),i\neq s} d_k^+\left(\boldsymbol{x_j^i}\right) + d_k^+\left(\boldsymbol{x_j^t}\right) - c_a\sum_{i\in I(k)',i\neq t} d_k^+\left(\boldsymbol{x_j^i}\right) - c_a d_k^+\left(\boldsymbol{x_j^s}\right) \tag{10}$$

$$\sum_{i\in I(k)',i\neq t} d_{k+1}^-\left(\boldsymbol{x_j^i}\right) + d_{k+1}^-\left(\boldsymbol{x_j^s}\right) - c_b\sum_{i\in I(k),i\neq s} d_{k+1}^-\left(\boldsymbol{x_j^i}\right) - c_b d_{k+1}^-\left(\boldsymbol{x_j^t}\right) + \gamma_4 = \sum_{i\in I(k)',i\neq t} d_k^-\left(\boldsymbol{x_j^i}\right) + d_k^-\left(\boldsymbol{x_j^s}\right) - c_b\sum_{i\in I(k),i\neq s} d_k^-\left(\boldsymbol{x_j^i}\right) - c_b d_k^-\left(\boldsymbol{x_j^t}\right). \tag{11}$$

Here, $\gamma_i \geq 0, i = 1, 2, 3, 4$.

Lemma 2: For a bag j, let s in $+1$ and t in -1 be the only switch pair. If $c_a = c_b$, then, $d_k^+(\boldsymbol{x_j^t}) + d_k^-(\boldsymbol{x_j^s}) - d_k^+(\boldsymbol{x_j^s}) - d_k^-(\boldsymbol{x_j^t}) \geq d_{k+1}^+(\boldsymbol{x_j^t}) + d_{k+1}^-(\boldsymbol{x_j^s}) - d_{k+1}^+(\boldsymbol{x_j^s}) - d_{k+1}^-(\boldsymbol{x_j^t})$.

Proof: Because s in $+1$ and t in -1 is the only switch pair, it follows $d^-(\boldsymbol{x_j^s}) - d^+(\boldsymbol{x_j^s}) < d^-(\boldsymbol{x_j^t}) - d^+(\boldsymbol{x_j^t})$.

By adding (8)–(11) together, we have

$$\begin{aligned} &d_{k+1}^+\left(\boldsymbol{x_j^t}\right) - d_{k+1}^+\left(\boldsymbol{x_j^s}\right) + d_{k+1}^-\left(\boldsymbol{x_j^s}\right) - d_{k+1}^-\left(\boldsymbol{x_j^t}\right) + \frac{\gamma_1+\gamma_2+\gamma_3+\gamma_4}{1+c_a} \\ &= d_k^+\left(\boldsymbol{x_j^t}\right) - d_k^+\left(\boldsymbol{x_j^s}\right) + d_k^-\left(\boldsymbol{x_j^s}\right) - d_k^-\left(\boldsymbol{x_j^t}\right) \end{aligned}$$

which means

$$\begin{aligned} &d_{k+1}^+\left(\boldsymbol{x_j^t}\right) - d_{k+1}^+\left(\boldsymbol{x_j^s}\right) + d_{k+1}^-\left(\boldsymbol{x_j^s}\right) - d_{k+1}^-\left(\boldsymbol{x_j^t}\right) \\ &\leq d_k^+\left(\boldsymbol{x_j^t}\right) - d_k^+\left(\boldsymbol{x_j^s}\right) + d_k^-\left(\boldsymbol{x_j^s}\right) - d_k^-\left(\boldsymbol{x_j^t}\right). \end{aligned} \tag{12}$$

■

Remark 1: Particularly, when there is some positive $\gamma_i, i = 1, 2, 3, 4$, $d_{k+1}^+(\boldsymbol{x_j^t}) - d_{k+1}^+(\boldsymbol{x_j^s}) + d_{k+1}^-(\boldsymbol{x_j^s}) - d_{k+1}^-(\boldsymbol{x_j^t}) < 0$ strictly holds. It follows that: in the $(k+1)$th iteration, $\boldsymbol{x_j^s}$ and $\boldsymbol{x_j^t}$ will not be a switch pair.

Definition 1: In iteration k, define a distance measure for $\boldsymbol{x}$ as follows: $m_k(\boldsymbol{x}) = d_k^-(\boldsymbol{x}) - d_k^+(\boldsymbol{x})$.

As we can see, the conclusion in Lemma 1 can be expressed by $m_{k+1}(\boldsymbol{x_j^s}) - m_{k+1}(\boldsymbol{x_j^t}) \leq m_k(\boldsymbol{x_j^s}) - m_k(\boldsymbol{x_j^t}) \leq 0$. Motivated by this conclusion, we build the following assumption.

Assumption 1: Let s and t be the positive and negative indexes switching in the label assignment procedure of the kth iteration. Assume that $m_r(\boldsymbol{x_j^s}) - m_r(\boldsymbol{x_j^t}) \leq 0$ with respect to $\forall r > k$.

Remark 2: If we regard m_k as a measure for the degree of a certain point belonging to positive class, In this way, Assumption 1 can be obtained by the following two more strict conditions:

$$m_r\left(\boldsymbol{x_j^s}\right) \leq m_k\left(\boldsymbol{x_j^s}\right), m_k\left(\boldsymbol{x_j^t}\right) \leq m_r\left(\boldsymbol{x_j^t}\right), \ \forall r > k. \tag{13}$$

However, we can tell that (13) is not a necessary condition for Assumption 1 in many real-life problems. That is to say, we give a much looser requirement for $m_k(\boldsymbol{x})$.

Remark 3: This assumption is not trivial. Particularly, $m_{k+1}(\boldsymbol{x_j^t}) < m_{k+1}(\boldsymbol{x_j^n})$ may hold, for some negative point $\boldsymbol{x_j^n}$ in the kth iteration. As a result, there is no guarantee for $\boldsymbol{x_j^t}$ not to be relabeled as a negative point in the following iterations.

Theorem 1: Based on Assumption 1 and Algorithm 1 is finite terminative.

Proof: First, according to Lemma 2, the switch is irreversible between two sequent iterations. Second, based on

Assumption 1, $m_k(\boldsymbol{x}_j^s) - m_k(\boldsymbol{x}_j^t) \leq 0$ with respect to $\forall r > k$. This guarantees that $\boldsymbol{x}_j^s$ and $\boldsymbol{x}_j^t$ will not be a switch pair in all the following iterations of the kth one. Because the number of points in every single bag is finite, Algorithm 1 is finite terminative. ■

However, noting Assumption 1 drastically depends on the distribution of data and may not be guaranteed in some extreme cases, which can hardly lead to finite termination, we would like to introduce the monotonic decrease with respect to objective function in Algorithm 1 instead.

C. Monotonic Decrease of Algorithm 1

For sake of simplicity, we only consider two principal terms to form the following objective function Obj^b, by eliminating the first regularization term corresponding to RKHS and the influence of ε-band associating to sparsity. This setting gives rise to an essential expression of our two-step model and can be cast handily in our discussion

$$\text{Obj}_k^b = \sum_{j=1}^{m} \left\{ \sum_{i\in I(k)} \left[d_k^+\left(\boldsymbol{x}_j^i\right) - c_a d_k^-\left(\boldsymbol{x}_j^i\right) \right] + \sum_{i\in I(k)'} \left[d_k^-\left(\boldsymbol{x}_j^i\right) - c_b d_k^+\left(\boldsymbol{x}_j^i\right) \right] \right\}. \tag{14}$$

As we can see, when parameters c_a and c_b are very close to zero, the effect of far away term cannot succeed much, which gives rise to rigorous oppression to the objective function. Consequently, a strict value decrease occurs. Here, we meticulously study the performance of the objective function with respect to very small positive parameters.

Theorem 2: Let $c_a = c_b$ and be small enough positive numbers. The objective function Obj^b decreases monotonously with respect to Algorithm 1.

Proof: In Algorithm 1, there are two steps in each iteration. Here, we specifically consider the kth iteration. For the trivial situation, if there is no switch occurs at all, the objective function will not change and algorithm stops. Otherwise, for simplicity, we consider only one pair switch occurs. For the first step, i.e., in cluster assignation, according to Lemma 1, the objective function Obj^b decreases. In the second step, i.e., cluster update, we have (10) and (11).

From (10) + (11), we obtain

$$\begin{aligned}
&\sum_{i\in I(k),i\neq s} d_{k+1}^+\left(\boldsymbol{x}_j^i\right) + d_{k+1}^+\left(\boldsymbol{x}_j^t\right) + \sum_{i\in I(k)',i\neq t} d_{k+1}^-\left(\boldsymbol{x}_j^i\right) + d_{k+1}^-\left(\boldsymbol{x}_j^s\right) \\
&= \sum_{i\in I(k),i\neq s} d_k^+\left(\boldsymbol{x}_j^i\right) + d_k^+\left(\boldsymbol{x}_j^t\right) \\
&\quad + \sum_{i\in I(k)',i\neq t} d_k^-\left(\boldsymbol{x}_j^i\right) + d_k^-\left(\boldsymbol{x}_j^s\right) \\
&\quad + c_a \sum_{i\in I(k)',i\neq t} \left(d_{k+1}^+\left(\boldsymbol{x}_j^i\right) - d_k^+\left(\boldsymbol{x}_j^i\right) \right) \\
&\quad + c_a \left(d_{k+1}^+\left(\boldsymbol{x}_j^s\right) - d_k^+\left(\boldsymbol{x}_j^s\right) \right) \\
&\quad + c_b \sum_{i\in I(k),i\neq s} \left(d_{k+1}^-\left(\boldsymbol{x}_j^i\right) - d_k^-\left(\boldsymbol{x}_j^i\right) \right) \\
&\quad + c_b \left(d_{k+1}^-\left(\boldsymbol{x}_j^t\right) - d_k^-\left(\boldsymbol{x}_j^t\right) \right) - \gamma_3 - \gamma_4.
\end{aligned} \tag{15}$$

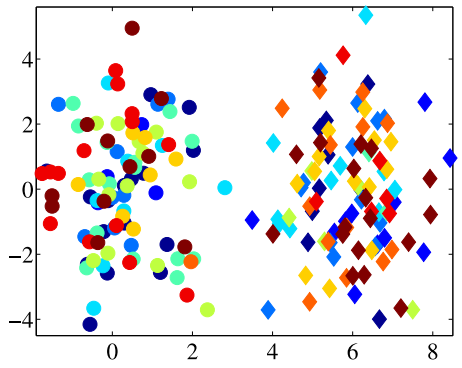


Fig. 3. Toy data. The number of each class is 100. The ∘ denotes the positive class and the ◊ denotes the negative class. There are ten bags in all, and the different colors denote different bags.

Plugging (8), (9) into (15), we obtain

$$\begin{aligned}
&\sum_{i\in I(k),i\neq s} d_{k+1}^+\left(\boldsymbol{x}_j^i\right) + d_{k+1}^+\left(\boldsymbol{x}_j^t\right) + \sum_{i\in I(k)',i\neq t} d_{k+1}^-\left(\boldsymbol{x}_j^i\right) + d_{k+1}^-\left(\boldsymbol{x}_j^s\right) \\
&= \sum_{i\in I(k),i\neq s} d_k^+\left(\boldsymbol{x}_j^i\right) + d_k^+\left(\boldsymbol{x}_j^t\right) \\
&\quad + \sum_{i\in I(k)',i\neq t} d_k^-\left(\boldsymbol{x}_j^i\right) + d_k^-\left(\boldsymbol{x}_j^s\right) \\
&\quad + c_a \sum_{i\in I(k)} \left(d_{k+1}^+\left(\boldsymbol{x}_j^i\right) - d_k^+\left(\boldsymbol{x}_j^i\right) \right) \\
&\quad + \gamma_1 + c_a \sum_{i\in I(k)'} \left(d_{k+1}^-\left(\boldsymbol{x}_j^i\right) - d_k^-\left(\boldsymbol{x}_j^i\right) \right) + \gamma_2 \\
&\quad + c_a \left[\left(m_{k+1}\left(\boldsymbol{x}_j^t\right) - m_{k+1}\left(\boldsymbol{x}_j^s\right) \right) - \left(m_k\left(\boldsymbol{x}_j^t\right) - m_k\left(\boldsymbol{x}_j^s\right) \right) \right] \\
&\quad - \gamma_1 - \gamma_2 - \gamma_3 - \gamma_4.
\end{aligned} \tag{16}$$

Again, plugging Section III-B into (16) leads to

$$\begin{aligned}
&\sum_{i\in I(k),i\neq s} d_{k+1}^+\left(\boldsymbol{x}_j^i\right) + d_{k+1}^+\left(\boldsymbol{x}_j^t\right) + \sum_{i\in I(k)',i\neq t} d_{k+1}^-\left(\boldsymbol{x}_j^i\right) + d_{k+1}^-\left(\boldsymbol{x}_j^s\right) \\
&= \sum_{i\in I(k),i\neq s} d_k^+\left(\boldsymbol{x}_j^i\right) + \sum_{i\in I(k)',i\neq t} d_k^-\left(\boldsymbol{x}_j^i\right) \\
&\quad + d_k^+\left(\boldsymbol{x}_j^t\right) + d_k^-\left(\boldsymbol{x}_j^s\right) + c_a \sum_{i\in I(k)} \left(d_{k+1}^+\left(\boldsymbol{x}_j^i\right) - d_k^+\left(\boldsymbol{x}_j^i\right) \right) \\
&\quad + c_a \sum_{i\in I(k)'} \left(d_{k+1}^-\left(\boldsymbol{x}_j^i\right) - d_k^-\left(\boldsymbol{x}_j^i\right) \right) \\
&\quad + \frac{c_a(\gamma_1+\gamma_2)}{1+c_a} - \frac{\gamma_3+\gamma_4}{1+c_a}.
\end{aligned} \tag{17}$$

Then, let $c_a, c_b \to 0+$, we have

$$\begin{aligned}
&\sum_{i\in I(k),i\neq s} d_{k+1}^+\left(\boldsymbol{x}_j^i\right) + d_{k+1}^+\left(\boldsymbol{x}_j^t\right) + \sum_{i\in I(k)',i\neq t} d_{k+1}^-\left(\boldsymbol{x}_j^i\right) + d_{k+1}^-\left(\boldsymbol{x}_j^s\right) \\
&= \sum_{i\in I(k),i\neq s} d_k^+\left(\boldsymbol{x}_j^i\right) + \sum_{i\in I(k)',i\neq t} d_k^-\left(\boldsymbol{x}_j^i\right) \\
&\quad + d_k^+\left(\boldsymbol{x}_j^t\right) + d_k^-\left(\boldsymbol{x}_j^s\right) - (\gamma_3+\gamma_4) < \sum_{i\in I(k),i\neq s} d_k^+\left(\boldsymbol{x}_j^i\right) \\
&\quad + \sum_{i\in I(k)',i\neq t} d_k^-\left(\boldsymbol{x}_j^i\right) + d_k^+\left(\boldsymbol{x}_j^t\right) + d_k^-\left(\boldsymbol{x}_j^s\right).
\end{aligned} \tag{18}$$

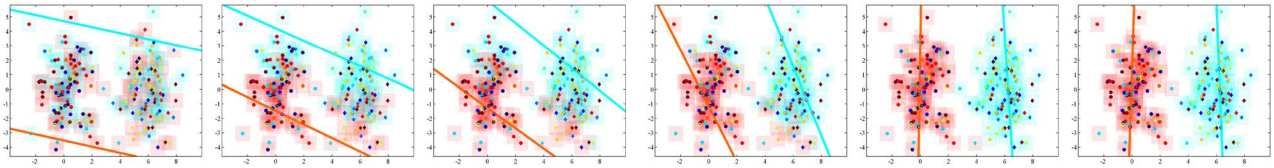


Fig. 4. Result of LLP-NPSVM in the toy data. The red and cyan lines are two hyper-planes obtained by the LLP-NPSVM, the red or cyan shadow of each point indicates the predicted result in each step.

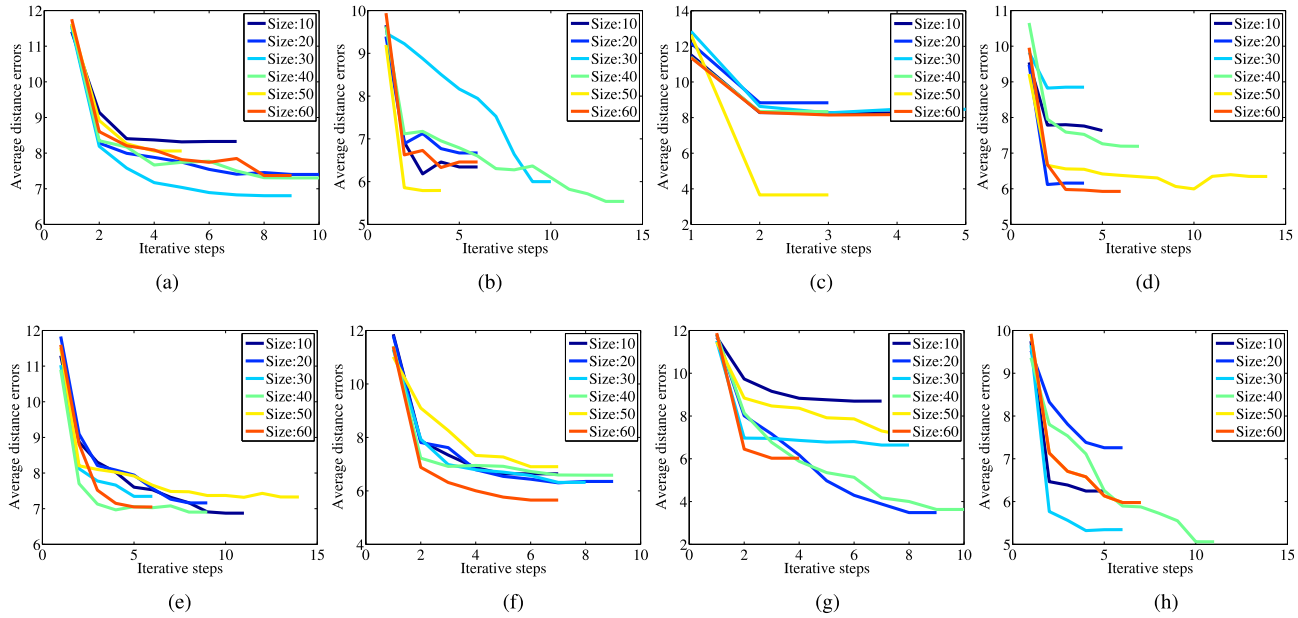


Fig. 5. LLP-NPSVM: the change of the average distance from the samples to hyper-planes with the increase of the number of iterations in the eight data sets. The different colors denote the results of different sizes of bags. (a) German-bank. (b) Breast-cancer. (c) Car. (d) Haberman. (e) Heart-statlog. (f) Ionosphere. (g) Letter. (h) Vowel.

Noting $d_k^-(\boldsymbol{x}_j^t) - d_k^+(\boldsymbol{x}_j^t) > d_k^-(\boldsymbol{x}_j^s) - d_k^+(\boldsymbol{x}_j^s)$, combining the above two results and (16), we can obtain the following result:

$$\sum_{i\in I(k), i\neq s} d_{k+1}^+\left(\boldsymbol{x}_j^i\right) + \sum_{i\in I(k)', i\neq t} d_{k+1}^-\left(\boldsymbol{x}_j^i\right) + d_{k+1}^+\left(\boldsymbol{x}_j^t\right) + d_{k+1}^-\left(\boldsymbol{x}_j^s\right) < \sum_{i\in I(k), i\neq s} d_k^+\left(\boldsymbol{x}_j^i\right) + \sum_{i\in I(k)', i\neq t} d_k^-\left(\boldsymbol{x}_j^i\right) + d_k^+\left(\boldsymbol{x}_j^s\right) + d_k^-\left(\boldsymbol{x}_j^t\right). \tag{19}$$

■

Remark 4: In fact, we have not only proved the strictly decrease between two label updates in the sequent iterations, but also shown the decrease between two steps in one iteration. Finally, in our algorithm, when the change of objective function is smaller than some predefined threshold, we can obtain an effective solution $(\boldsymbol{w}^*, b^*)$ for the problem in the corresponding label update step.

IV. Experiment

Running environment: MATLAB 2010 on a PC with an Intel Core I5 processor and 4 GB RAM.

A. How Does the LLP-NPSVM Work?

First, we would like to give a simple example on the toy data set in Fig. 3, which is generated by two 2-variate normal distributions with parameters: 1) $\mu_1 = (0, 0)$, $\sigma_1 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$ and 2) $\mu_2 = (3, 0)$, $\sigma_2 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$, respectively. There are 100 points in each class. Particularly, The "$\circ$" denotes the positive class and the "$\diamondsuit$" denotes the negative class. There are ten bags in total, and points in different bags are marked in different colors. This experiment is aimed to intuitively illustrate how LLP-NPSVM works. In order to fully present the behavior of iterative process, we deliberately choose a relative "bad" initiation for $\boldsymbol{y}$. The results are displayed in Fig. 4 stepwise.

As is seen in Fig. 4, the red and cyan lines are two hyper-planes obtained by the LLP-NPSVM stepwise. In the meantime, red and cyan shadows indicate the prediction results of corresponding points in each step. From Fig. 4, we can tell that our algorithm only takes six steps to achieve stable result, that is, all samples are correctly classified. Furthermore, if we replace the deliberately bad initiation $\boldsymbol{y}$ with a random selection, this process can be stabilized within three iterations on average. From the change of the samples' shadows, we can easily perceive the convergence process of the algorithm. That is to say, those shadows with the same color are aggregated

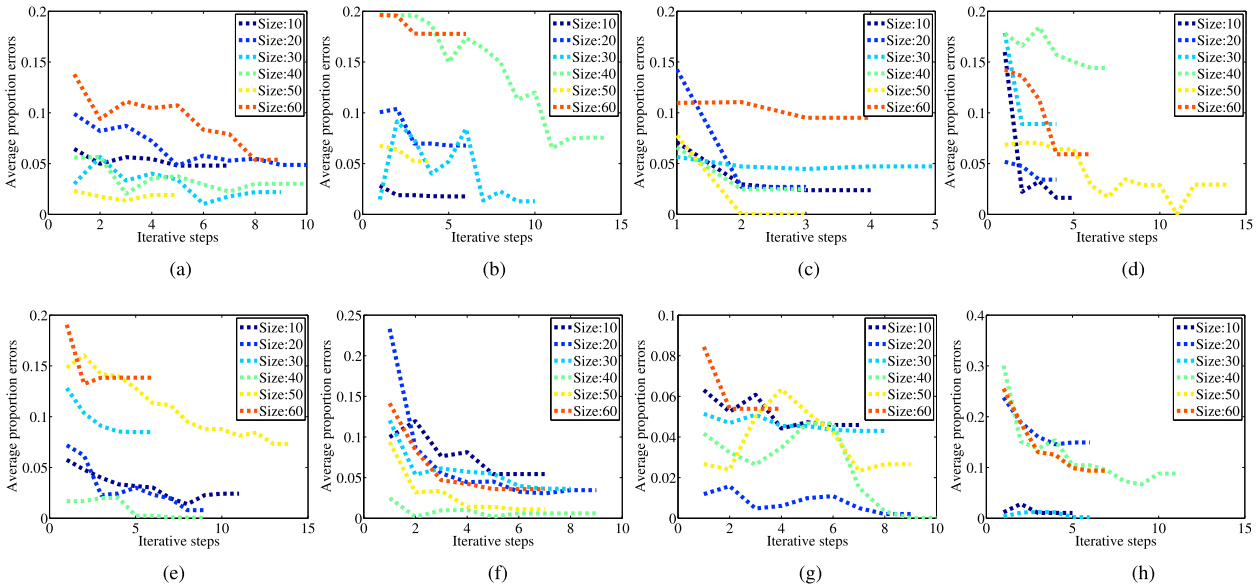


Fig. 6. LLP-NPSVM: the change of the average proportion error with the increase of the number of iterations in the eight data sets. The different colors denote the results of different sizes of bags. (a) German-bank. (b) Breast-cancer. (c) Car. (d) Haberman. (e) Heart-statlog. (f) Ionosphere. (g) Letter. (h) Vowel.

TABLE I
SELECTED UCI DATA SETS

Dataset	Size	Attributes	Classes
german-bank	1000	20	2
breast-cancer	277	10	2
car	1594	6	4 (1,2)
haberman	306	3	2
heart-statlog	270	13	2
ionosphere	351	34	2
letter	1555	16	26 (a,b)
vowel	180	10	11 (1,2)

These data sets have been reorganized, where $(\cdot, \cdot)$ denotes the selected classes in the multi-class classification. More details can be found in https://github.com/qizhiquan/LLP-NPSVM.

continuously and smoothly. In the last figure, this method can clearly attain the optimal solution, that is, all positive points are covered by the shadows with red color, meanwhile, all the negative points are covered by the shadows with cyan color.

B. UCI Datasets for Binary-Class Problem

In this section, the proposed algorithm is applied to UCI repository data sets[1] to evaluate its effectiveness, compared to InvCal [26], conv-$\propto$SVM [27], and alter-$\propto$SVM [27]. The codes of these methods are available in https://github.com/felixyu/pSVM. Meanwhile, our code can be found in https://github.com/qizhiquan/LLP-NPSVM. Table I gives overall description of the selected UCI data sets.

Our experiment only accounts for binary classification situation. For multiclass scenario, 1-versus-1 strategy is performed based on binary classification. In bag setting, a random selection is applied to the original data set, with varied sizes ranged in 10, 20, 30, 40, 50, and 60. Then, 80% bags are used for training, and the rest for test. The average classification accuracies with standard deviations can be obtained by repeating the above process five times. All the parameters are tuned by the

[1]http://archive.ics.uci.edu/ml/

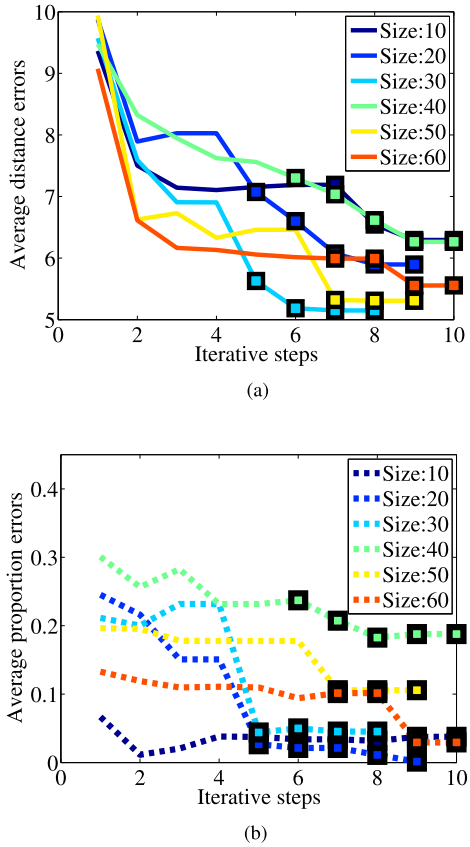


Fig. 7. Change of both the average distance and the average proportion before and after the switch strategy in LLP-NPSVM, the broken lines before without □ denote the results before swapping models, and the broken lines with □ denote the results after swapping models. (a) Change of the average distance in the breast-cancer data set. (b) Change of the average proportion in the breast-cancer data set.

fivefold cross validation on the training subsets. Also, the error rates of training data are evaluated by the measure of bag-level error: $\text{Err} = \sum_{i=1}^{l} \|\tilde{p}_i - p_i\|$, where $\tilde{p}_i$ and p_i are the predicted

TABLE II
FIVEFOLD CROSS VALIDATION RESULTS FOR BINARY CLASSIFICATION WITH LINEAR KERNEL

Dataset	Method	10	20	30	40	50	60
german-bank	InvCal	0.69 ± 0.02	0.58 ± 0.01	0.65 ± 0.01	0.64 ± 0.01	0.54 ± 0.02	0.63 ± 0.01
	conv∝SVM	0.57 ± 0.01	0.50 ± 0.00	0.50 ± 0.02	0.57 ± 0.01	0.51 ± 0.01	0.60 ± 0.01
	alter-∝SVM	0.60 ± 0.01	0.54 ± 0.02	0.62 ± 0.01	0.62 ± 0.02	0.57 ± 0.00	0.53 ± 0.01
	LLP-NPSVM	0.71± 0.01	0.62 ± 0.01	0.62 ± 0.01	0.65 ± 0.01	0.52 ± 0.03	0.51 ± 0.04
breast-cancer	InvCal	0.62 ± 0.01	0.59 ± 0.00	0.49 ± 0.02	0.55 ± 0.02	0.58 ± 0.00	0.54 ± 0.01
	conv∝SVM	0.58 ± 0.01	0.57 ± 0.01	0.55 ± 0.01	0.57 ± 0.01	0.54 ± 0.01	0.52 ± 0.01
	alter-∝SVM	0.65 ± 0.01	0.59 ± 0.04	0.62 ± 0.01	0.53 ± 0.02	0.49 ± 0.02	0.46 ± 0.02
	LLP-NPSVM	0.67 ± 0.03	0.61 ± 0.01	0.61 ± 0.02	0.58 ± 0.02	0.52 ± 0.03	0.51 ± 0.02
car	InvCal	0.98 ± 0.02	0.98 ± 0.01	0.95 ± 0.01	0.93 ± 0.01	0.90 ± 0.00	0.89 ± 0.01
	conv∝SVM	0.98 ± 0.01	0.94 ± 0.16	0.93 ± 0.11	0.93 ± 0.24	0.91 ± 0.04	0.90 ± 0.01
	alter-∝SVM	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
	LLP-NPSVM	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
haberman	InvCal	0.75 ± 0.04	0.66 ± 0.05	0.71 ± 0.03	0.62 ± 0.01	0.82 ± 0.01	0.82 ± 0.01
	conv∝SVM	0.74 ± 0.01	0.74 ± 0.01	0.67 ± 0.01	0.62 ± 0.01	0.61 ± 0.01	0.54 ± 0.02
	alter-∝SVM	0.79 ± 0.02	0.82 ± 0.01	0.74 ± 0.03	0.79 ± 0.01	0.71 ± 0.03	0.66 ± 0.01
	LLP-NPSVM	0.84 ± 0.00	0.81 ± 0.00	0.76 ± 0.02	0.75 ± 0.02	0.72 ± 0.02	0.61 ± 0.02
heart-statlog	InvCal	0.70 ± 0.00	0.69 ± 0.01	0.68 ± 0.01	0.64 ± 0.01	0.68 ± 0.01	0.66 ± 0.00
	conv∝SVM	0.65 ± 0.01	0.60 ± 0.03	0.60 ± 0.01	0.59 ± 0.01	0.54 ± 0.01	0.56 ± 0.01
	alter-∝SVM	0.76 ± 0.00	0.75 ± 0.01	0.68 ± 0.02	0.76 ± 0.01	0.68 ± 0.02	0.68 ± 0.01
	LLP-NPSVM	0.75 ± 0.01	0.74 ± 0.01	0.71 ± 0.01	0.72 ± 0.01	0.71 ± 0.02	0.70 ± 0.02
ionosphere	InvCal	0.80 ± 0.00	0.74 ± 0.00	0.72 ± 0.00	0.74 ± 0.01	0.64 ± 0.01	0.64 ± 0.00
	conv∝SVM	0.78 ± 0.01	0.75 ± 0.02	0.74 ± 0.01	0.68 ± 0.01	0.65 ± 0.01	0.71 ± 0.01
	alter-∝SVM	0.77 ± 0.00	0.75 ± 0.01	0.75 ± 0.01	0.73 ± 0.01	0.73 ± 0.01	0.67 ± 0.01
	LLP-NPSVM	0.75 ± 0.01	0.76 ± 0.01	0.74 ± 0.01	0.72 ± 0.01	0.74 ± 0.02	0.65 ± 0.02
letter	InvCal	0.94 ± 0.00	0.91 ± 0.01	0.93 ± 0.01	0.95 ± 0.01	0.89 ± 0.01	0.91 ± 0.00
	conv∝SVM	0.89 ± 0.01	0.87 ± 0.01	0.85 ± 0.01	0.85 ± 0.01	0.87 ± 0.01	0.81 ± 0.01
	alter-∝SVM	0.98 ± 0.00	0.94 ± 0.02	0.93 ± 0.01	0.93 ± 0.01	0.89 ± 0.02	0.76 ± 0.03
	LLP-NPSVM	0.99 ± 0.01	0.99 ± 0.01	0.99 ± 0.01	0.97 ± 0.01	0.96 ± 0.02	0.96 ± 0.01
vowel	InvCal	0.71 ± 0.00	0.57 ± 0.01	0.52 ± 0.01	0.53 ± 0.00	0.51 ± 0.01	0.51 ± 0.00
	conv∝SVM	0.68 ± 0.01	0.67 ± 0.01	0.61 ± 0.01	0.58 ± 0.01	0.54 ± 0.01	0.52 ± 0.01
	alter-∝SVM	0.67 ± 0.01	0.63 ± 0.01	0.59 ± 0.01	0.59 ± 0.01	0.55 ± 0.01	0.56 ± 0.01
	LLP-NPSVM	0.74 ± 0.01	0.68 ± 0.02	0.66 ± 0.02	0.63 ± 0.02	0.62 ± 0.02	0.60 ± 0.01

The LLP-NPSVM's results are marked in grape when it outperforms all the other methods (InvCal, alter-∝SVM and conv-∝SVM). Otherwise, the best results are marked in pale green.

and ground-truth proportions for the ith validation bag [27], respectively. The parameter settings are shown as follows.

InvCal: $\lambda \in [0.1, 1, 10]$, $C_p \in [0.1, 1, 10]$, $\varepsilon \in [0, 0.01, 0.1]$.

alter-∝SVM: $C \in [0.1, 1, 10]$, $C_p \in [1, 10, 100]$.

conv-∝SVM: $C \in [0.1, 1, 10]$, $\varepsilon \in [0, 0.01, 0.1]$.

LLP-NPSVM: $C \in [0.01, 0.1, 1]$, $\varepsilon \in [0.001, 0.01, 0.1]$.

Besides, in label update with RBF kernel: γ is unified to $[0.01, 0.1, 1]$.

Fig. 5 manifestly demonstrates LLP-NPSVM's the change of average distances from the samples to hyper-planes in the eight data sets. Obviously, almost all average distances under the different sizes of bags are monotonously decreasing with the increase of iteration number, which intuitively reveals the convergence of the LLP-NPSVM. In addition, it can be also observed that most of the average errors attain stable within 5 iterations, which fully supports the conclusion that our algorithm has a character of fast convergence. Synchronously, Fig. 6 displays the change of the average proportion error in each iteration. Compared with Fig. 5, we do not obtain an approximately convergent result similar to that of the average distances. Especially, the average proportion errors cannot achieve a strict decline in certain sizes of bags (see the yellow line in the bag's size of 50 in "letter" data set for example), because the goal of LLP-NPSVM is to minimize the average distance instead of the average proportion error. Although we expect average proportion error declines during the iterative process in the algorithm, there is no rigorous requirement for monotonously dropping. Nevertheless, from the result of Fig. 5, most average proportion errors decline along with the increase of iteration number, which satisfies the demand of our implicity expectation in LLP and is committed to be an excellent result. Combining Figs. 5 and 6, in major situation, both the average distances and average proportion errors are simultaneously decreasing within few iterations, which bears a satisfactory convergence for LLP-NPSVM.

On the other hand, Fig. 7 captures the results before and after the switch strategy (marked in Algorithm 1) on "breast-cancer" data set. This experiment is supposed to demonstrate the effectiveness of this swapping trick in LLP-NPSVM. Particularly, the lines without "□" denote the results before the swapping procedure. Meanwhile, the lines marked with □ denote the results after the swapping procedure. Based on Fig. 7, we can tell that both the average distances and average proportion errors are provoked a further decline in most cases. In other words, the swapping tactics can alleviate the unwilling attraction to local optima and obtain a better solution for LLP-NPSVM somehow.

Additionally, Tables II and III display the fivefold cross validation results of InvCal, conv-∝SVM, alter-∝SVM, and LLP-NPSVM for binary classification problems. First, in Table II, LLP-NPSVM overwhelmingly wins under linear kernel, i.e., 35 best results out of total 48 results.

TABLE III
FIVEFOLD CROSS VALIDATION RESULTS FOR BINARY CLASSIFICATION WITH RBF KERNEL

Dataset	Method	10	20	30	40	50	60
german-bank	InvCal	0.74 ± 0.01	0.74 ± 0.01	0.70 ± 0.00	0.70 ± 0.01	0.60 ± 0.01	0.60 ± 0.01
	conv-∝SVM	0.74 ± 0.02	0.66 ± 0.00	0.66 ± 0.01	0.63 ± 0.01	0.69 ± 0.00	0.65 ± 0.01
	alter-∝SVM	0.71 ± 0.02	0.71 ± 0.02	0.69 ± 0.01	0.68 ± 0.00	0.65 ± 0.01	0.54 ± 0.02
	LLP-NPSVM	0.74 ± 0.02	0.75 ± 0.01	0.78 ± 0.01	0.75 ± 0.01	0.79 ± 0.00	0.62 ± 0.02
breast-cancer	InvCal	0.61 ± 0.01	0.55 ± 0.01	0.50 ± 0.01	0.53 ± 0.01	0.55 ± 0.01	0.53 ± 0.01
	conv-∝SVM	0.64 ± 0.01	0.61 ± 0.00	0.56 ± 0.02	0.54 ± 0.02	0.56 ± 0.01	0.61 ± 0.00
	alter-∝SVM	0.65 ± 0.02	0.64 ± 0.01	0.61 ± 0.01	0.58 ± 0.02	0.47 ± 0.01	0.62 ± 0.01
	LLP-NPSVM	0.69 ± 0.01	0.67 ± 0.01	0.61 ± 0.00	0.55 ± 0.00	0.53 ± 0.02	0.63 ± 0.02
car	InvCal	0.98 ± 0.02	0.96 ± 0.02	0.93 ± 0.01	0.98 ± 0.01	0.77 ± 0.01	0.88 ± 0.01
	conv-∝SVM	0.98 ± 0.02	0.95 ± 0.01	0.94 ± 0.00	0.93 ± 0.01	0.92 ± 0.01	0.91 ± 0.02
	alter-∝SVM	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
	LLP-NPSVM	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
haberman	InvCal	0.86 ± 0.01	0.75 ± 0.02	0.69 ± 0.01	0.64 ± 0.01	0.60 ± 0.01	0.64 ± 0.01
	conv-∝SVM	0.85 ± 0.02	0.73 ± 0.02	0.66 ± 0.01	0.69 ± 0.01	0.60 ± 0.01	0.61 ± 0.01
	alter-∝SVM	0.88 ± 0.00	0.78 ± 0.01	0.76 ± 0.02	0.74 ± 0.00	0.62 ± 0.01	0.61 ± 0.02
	LLP-NPSVM	0.89 ± 0.01	0.75 ± 0.01	0.77 ± 0.01	0.76 ± 0.01	0.66 ± 0.01	0.74 ± 0.01
heart-statlog	InvCal	0.76 ± 0.02	0.71 ± 0.01	0.70 ± 0.00	0.62 ± 0.01	0.60 ± 0.02	0.60 ± 0.01
	conv-∝SVM	0.76 ± 0.01	0.72 ± 0.01	0.68 ± 0.02	0.71 ± 0.01	0.64 ± 0.01	0.62 ± 0.01
	alter-∝SVM	0.74 ± 0.02	0.70 ± 0.02	0.70 ± 0.01	0.68 ± 0.01	0.63 ± 0.01	0.65 ± 0.01
	LLP-NPSVM	0.79 ± 0.01	0.76 ± 0.01	0.71 ± 0.00	0.79 ± 0.01	0.72 ± 0.01	0.71 ± 0.02
ionosphere	InvCal	0.79 ± 0.02	0.75 ± 0.01	0.61 ± 0.01	0.59 ± 0.02	0.61 ± 0.00	0.50 ± 0.01
	conv-∝SVM	0.76 ± 0.01	0.76 ± 0.01	0.67 ± 0.01	0.66 ± 0.01	0.64 ± 0.01	0.64 ± 0.01
	alter-∝SVM	0.78 ± 0.02	0.76 ± 0.02	0.68 ± 0.01	0.67 ± 0.01	0.64 ± 0.00	0.61 ± 0.02
	LLP-NPSVM	0.81 ± 0.01	0.77 ± 0.00	0.77 ± 0.01	0.76 ± 0.00	0.76 ± 0.01	0.75 ± 0.01
letter	InvCal	0.87 ± 0.02	0.84 ± 0.01	0.63 ± 0.01	0.72 ± 0.01	0.64 ± 0.01	0.64 ± 0.00
	conv-∝SVM	0.44 ± 0.01	0.74 ± 0.01	0.55 ± 0.02	0.45 ± 0.01	0.57 ± 0.01	0.40 ± 0.01
	alter-∝SVM	0.95 ± 0.01	0.94 ± 0.01	0.91 ± 0.01	0.93 ± 0.02	0.90 ± 0.02	0.94 ± 0.01
	LLP-NPSVM	0.99 ± 0.01	0.99 ± 0.01	0.98 ± 0.02	0.97 ± 0.01	0.96 ± 0.02	0.95 ± 0.01
vowel	InvCal	0.74 ± 0.01	0.73 ± 0.01	0.72 ± 0.02	0.66 ± 0.01	0.63 ± 0.01	0.61 ± 0.02
	conv-∝SVM	0.74 ± 0.02	0.72 ± 0.01	0.70 ± 0.01	0.64 ± 0.01	0.61 ± 0.01	0.59 ± 0.01
	alter-∝SVM	0.75 ± 0.00	0.72 ± 0.02	0.68 ± 0.00	0.72 ± 0.01	0.68 ± 0.02	0.62 ± 0.01
	LLP-NPSVM	0.76 ± 0.01	0.74 ± 0.01	0.71 ± 0.01	0.74 ± 0.01	0.71 ± 0.01	0.64 ± 0.02

The LLP-NPSVM's results are marked in grape when it outperforms all the other methods (InvCal, alter-∝SVM and conv-∝SVM). Otherwise, the best results are marked in pale green.

TABLE IV
WILCOXON SIGNED RANKS TEST RESULTS FOR BINARY CLASSIFICATION

	Linear Kernel			RBF Kernel		
	R^+	R^-	p-value	R^+	R^-	p-value
LLP-NPSVM VS. InvCal	952	224	0.00018	1070	106	0.00000
LLP-NPSVM VS. alter-∝SVM	1069.5	106.5	0.00000	1107	69	0.00000
LLP-NPSVM VS. conv-∝SVM	678	498	0.00129	831	345	0.00000

TABLE V
SOME EXAMPLES OF THE ATTRIBUTES

Name	Description and Values
Age	Age at the contact date (Numeric ≥18)
Job	Unemployed, Management, Housemaid, Entrepreneur, Student, Blue-collar, Self-employed, Retired... (Categorical)
Marital Status	Married, Single, Divorced, Widowed, Separated (Nominal)
Sex	Male or Female (Nominal)
Education	Secondary, Primary, Tertiary (Categorical)
Credit in default?	Yes or No (Nominal)
Annual Balance	in euro currency (Numeric)
Housing Load	Yes or No (Nominal)
Debt card?	Yes or No (Nominal)
Loans in delay?	Yes or No (Nominal)

TABLE VI
RESULTS ON THE CASE STUDY: PRIVACY-PRESERVING DATA MINING

Method	Job	Marital Status	Education
InvCal	0.79 ± 0.01	0.74 ± 0.03	0.69 ± 0.02
conv-∝SVM	0.81 ± 0.02	0.76 ± 0.03	0.67 ± 0.01
alter-∝SVM	0.82 ± 0.02	0.77 ± 0.05	0.72 ± 0.01
LLP-NPSVM	0.83± 0.01	0.79 ± 0.01	0.74 ± 0.03

In other words, based on our experiments, LLP-NPSVM dominates to yield better results with 0.7292 empirical probability. Especially on the letter and the "vowel" data sets,

TABLE VII
MULTICLASS UCI DATASETS

Dataset	Size	Attributes	Classes
acoustic	78823	50	3
combined	78823	100	3
connect-4	67557	126	3
covtype	581012	54	7

LLP-NPSVM outperforms all the other methods. On the "car" data set, both the accuracies of LLP-NPSVM and alter-∝SVM reach 100% at the same time. For InvCal algorithm, in the case of bag size 60, it obtains a better performance. Besides, alter-∝SVM is sightly better than conv-∝SVM, and is superior to other methods. This result also indicates a promising performance of maximum margin method. However, when increasing the bag size, accuracies of all methods decrease in varying degrees, which manifests the bigger bag size leads to bigger challenge. Second, according to Table III, when it comes to RBF kernel situation, unlike it is shown in Table II, InvCal also offers a comparable performance with the bag size of 10 on "German-bank" data set. In addition, conv-∝SVM is

TABLE VIII
FIVEFOLD CROSS VALIDATION RESULTS FOR MULTICLASS CLASSIFICATION WITH RBF KERNEL

Dataset	Method	2	4	8	16	32	64
acoustic	InvCal	0.74 ± 0.02	0.71 ± 0.08	0.66 ± 0.01	0.13 ± 0.01	0.48 ± 0.23	0.47 ± 0.04
	conv-∝SVM	0.54 ± 0.09	0.46 ± 0.08	0.31 ± 0.05	0.28 ± 0.07	0.44 ± 0.06	0.36 ± 0.08
	alter-∝SVM	0.68± 0.04	0.67 ± 0.01	0.64 ± 0.04	0.65 ± 0.03	0.44 ± 0.03	0.39 ± 0.03
	LLP-NPSVM	0.81 ± 0.04	0.74 ± 0.02	0.67 ± 0.01	0.55 ± 0.02	0.49 ± 0.03	0.45 ± 0.02
combined	InvCal	0.78 ± 0.02	0.76 ± 0.02	0.73 ± 0.02	0.75 ± 0.07	0.63 ± 0.05	0.51 ± 0.12
	conv-∝SVM	0.56 ± 0.03	0.63 ± 0.10	0.47± 0.09	0.44 ± 0.06	0.43 ± 0.04	0.42 ± 0.03
	alter-∝SVM	0.75 ± 0.06	0.77 ± 0.04	0.69 ± 0.08	0.51 ± 0.01	0.46 ± 0.02	0.44 ± 0.09
	LLP-NPSVM	0.77 ± 0.02	0.75 ± 0.02	0.74 ± 0.01	0.73 ± 0.02	0.68 ± 0.02	0.59 ± 0.03
connect-4	InvCal	0.67 ± 0.06	0.60 ± 0.05	0.44 ± 0.05	0.38 ± 0.06	0.35 ± 0.07	0.35 ± 0.06
	conv-∝SVM	0.75 ± 0.04	0.71 ± 0.06	0.63± 0.06	0.63 ± 0.06	0.61 ± 0.08	0.43 ± 0.07
	alter-∝SVM	0.78 ± 0.09	0.45 ± 0.04	0.64 ± 0.09	0.66 ± 0.09	0.59 ± 0.02	0.55 ± 0.07
	LLP-NPSVM	0.81 ± 0.04	0.74 ± 0.06	0.68 ± 0.06	0.65 ± 0.06	0.63 ± 0.01	0.56 ± 0.02
covtype	InvCal	0.91 ± 0.05	0.87 ± 0.09	0.72 ± 0.03	0.61 ± 0.03	0.49 ± 0.08	0.29 ± 0.09
	conv-∝SVM	0.76 ± 0.04	0.73 ± 0.05	0.66± 0.08	0.54 ± 0.06	0.51 ± 0.08	0.32 ± 0.01
	alter-∝SVM	0.92 ± 0.12	0.86 ± 0.04	0.78 ± 0.07	0.64 ± 0.09	0.67 ± 0.01	0.41 ± 0.04
	LLP-NPSVM	0.89 ± 0.07	0.88 ± 0.03	0.81 ± 0.04	0.72 ± 0.04	0.66 ± 0.02	0.43 ± 0.07

The LLP-NPSVM's results are marked in grape when it outperforms all the other methods (InvCal, alter-∝SVM and conv-∝SVM). Otherwise, the best results are marked in pale green.

TABLE IX
WILCOXON SIGNED RANKS TEST RESULTS FOR MULTICLASS CLASSIFICATION

	RBF Kernel		
	R^+	R^-	p-value
LLP-NPSVM VS. InvCal	269	31	0.00066
LLP-NPSVM VS. alter-∝SVM	300	0	0.00002
LLP-NPSVM VS. conv-∝SVM	262	38	0.00140

able to attain better results in some case as well (see the first column in vowel data set). In spite of this, these two methods mentioned above cannot yield appropriately promising results compared to alter-∝SVM in most cases. Nevertheless, LLP-NPSVM is prone to acquire more competitive results than alter-∝SVM in all data sets. Wilcoxon signed ranks test results for binary classification are shown in Table IV. As we can see, the accuracies of LLP-NPSVM is significantly superior to InvCal, alter-∝SVM, and conv-∝SVM in binary classification problems. That is to say, besides similarly benefiting from maximum margin principle, LLP-NPSVM shows an overwhelming advantage thanks to the preservation of geometric distribution.

C. UCI Datasets for Multiclass Problem

In this section, the performance of LLP-NPSVM in dealing with multiclass classification problem will be evaluated. In detail, Table VII describes the multiclass datasets. Here, we apply 1-versus-rest strategy for all methods. We randomly sample 3000 points from each dataset for the training, and 600 points as the test data. Also, bags are randomly assembled from the selected data set, whose sizes are separately set to 2, 4, 8, 16, 32, and 64. Each experiment is repeated six times. Table VIII gives the final results.

In the total 24 experiments, our method wins 16 times, and is also the final winner based on overall evaluation. Besides, the alter-∝SVM obtains 11 winners; InvCal also obtains 11 winners; the conv-∝SVM performs the worst: only wins two times. In brief, these results demonstrate that InvCal achieves increasingly good performance with the increase of samples and classes. Wilcoxon signed ranks test results for multiclass classification are shown in Table IX. We can also learn that LLP-NPSVM has advantage in solving multiclass LLP problems.

D. Privacy-Preserving Data Mining Problem

In this section, a more challenging data set "privacy-preserving data mining" is used to evaluate and reveal the power of all the methods mentioned above in handling LLP problem. In this special case, our task is to predict whether the customer is willing to subscribe a term deposit according to the bank marketing data [44]. Explicitly, this data set contains 41 188 instances/records (individual persons) described by 20 attributes (including age, type of job, marital status, education, etc.). Table V shows some attributes of these customers. In common sense, each instance corresponds to a certain label in $\{1, -1\}$, which indicates the status of its term deposit subscribing. However, these label information are so sensitive due to the purpose of privacy protection that there is no visible instance-level label. As a compromise, let us assume that only the label proportions on different groups of instances are available in the training stage. These bags are grouped in three independent attributes: "job," "marital status," and "education." Table VI displays the final results. In practise, 80% of the instances are selected for training, and the rest for testing. Here, we adopt the bag error to evaluate these methods' performance. LLP-NPSVM gains the best performance in different partitioned bags, which demonstrates our method has a strong competitiveness in this alternative challenging situation under different privacy-preserving policies.

V. CONCLUSION

In this paper, we propose an EM-type algorithm for LLP problem. In our mechanism, label assignment and label update can be viewed as E-step and M-step, respectively, where maximum likelihood estimation in EM algorithm is coordinate to NPSVM in our method. The posterior probability of latent variable corresponds to distance in the clustering.

Particularly, we adopt k-plane clustering method to cope with intricate combinatorial optimization puzzle commonly encountering in weak label classification problems. To abate the influence of local optima, annealing loop is equipped. On one hand, with a substantial analysis on finite termination and monotonic decrease of objective function, we explore the stability of proposed method. On the other hand, meticulous and abundant experiments strongly verify and confirm those theoretical judgments. More importantly, it completely demonstrates the outstanding efficiency of LLP-NPSVM. In addition to kernel-based k-plane clustering, the outlook of the proposed method can be converted into metric learning regime or manifold-based clustering, for example adding Laplacian-like regularization term. In this way, structural information is fully revealed to offset the information loss with respect to weak label. To defeat the instability caused by label assignment and outliers, alternative convex loss function, for example pin-ball loss, can be employed to enhance the robustness.

ACKNOWLEDGMENT

The authors would like to thank Dr. F. X. Yu and Dr. S. Rueping for sharing their codes, which brought them great convenience for carrying out their experiments.

REFERENCES

[1] D. Kelly and B. Caulfield, "Pervasive sound sensing: A weakly supervised training approach," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 123–135, Jan. 2016.

[2] L. Zhang, X. Li, L. Nie, Y. Yang, and Y. Xia, "Weakly supervised human fixations prediction," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 258–269, Jan. 2016.

[3] Z.-G. Liu, Q. Pan, G. Mercier, and J. Dezert, "A new incomplete pattern classification method based on evidential reasoning," *IEEE Trans. Cybern.*, vol. 45, no. 4, pp. 635–646, Apr. 2015.

[4] G. Huang, S. Song, J. N. D. Gupta, and C. Wu, "Semi-supervised and unsupervised extreme learning machines," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2405–2417, Dec. 2014.

[5] W. Bian and X. Chen, "Neural network for nonsmooth, nonconvex constrained minimization via smooth approximation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 545–556, Mar. 2014.

[6] Z.-B. Xu, R. Zhang, and W.-F. Jing, "When does online BP training converge?" *IEEE Trans. Neural Netw.*, vol. 20, no. 10, pp. 1529–1539, Oct. 2009.

[7] D. Wang, J. Zhai, H. Zhu, and X. Wang, "An improved approach to ordinal classification," in *Machine Learning and Cybernetics*. Heidelberg, Germany: Springer, 2014, pp. 33–42.

[8] J. Yu and P. Hao, "Comments on 'the multisynapse neural network and its application to fuzzy clustering,'" *IEEE Trans. Neural Netw.*, vol. 16, no. 3, pp. 777–778, May 2005.

[9] F.-Y. Wang, N. Jin, D. Liu, and Q. Wei, "Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with-error bound," *IEEE Trans. Neural Netw.*, vol. 22, no. 1, pp. 24–36, Feb. 2011.

[10] Y.-H. Shao, C.-H. Zhang, X.-B. Wang, and N.-Y. Deng, "Improvements on twin support vector machines," *IEEE Trans. Neural Netw.*, vol. 22, no. 6, pp. 962–968, Jun. 2011.

[11] Z. Wang, Y.-H. Shao, L. Bai, and N.-Y. Deng, "Twin support vector machine for clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2583–2588, Oct. 2015.

[12] J. D. Keeler, D. E. Rumelhart, and W.-K. Leow, "Integrated segmentation and recognition of hand-printed numerals," in *Proc. Adv. Neural Inf. Process. Syst.*, 1991, pp. 557–563.

[13] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artif. Intell.*, vol. 89, nos. 1–2, pp. 31–71, 1997.

[14] A. Blum and A. Kalai, "A note on learning from multiple-instance examples," *Mach. Learn.*, vol. 30, no. 1, pp. 23–29, 1998.

[15] S. Sabato and N. Tishby, "Multi-instance learning with any hypothesis class," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 2999–3039, 2012.

[16] B. Babenko, N. Verma, P. Dollár, and S. J. Belongie, "Multiple instance learning with manifold bags," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, Bellevue, WA, USA, 2011, pp. 81–88.

[17] M.-L. Zhang and Z.-H. Zhou, "Improve multi-instance neural networks through feature selection," *Neural Process. Lett.*, vol. 19, no. 1, pp. 1–10, 2004.

[18] Z.-H. Zhou, M.-L. Zhang, S.-J. Huang, and Y.-F. Li, "Multi-instance multi-label learning," *Artif. Intell.*, vol. 176, no. 1, pp. 2291–2320, 2012.

[19] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 561–568.

[20] Z.-H. Zhou and J.-M. Xu, "On the relation between multi-instance learning and semi-supervised learning," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, Corvallis, OR, USA, 2007, pp. 1167–1174.

[21] A. J. Smola, S. V. N. Vishwanathan, and T. Hofmann, "Kernel methods for missing variables," in *Proc. 10th Int. Workshop Artif. Intell. Stat.*, Bridgetown, Barbados, 2005, pp. 325–332.

[22] H. Kuck and N. de Freitas, "Learning about individuals from group statistics," in *Proc. 21th UAI*, Edinburgh, U.K., 2005, pp. 332–339.

[23] B.-C. Chen, L. Chen, R. Ramakrishnan, and D. R. Musicant, "Learning from aggregate views," in *Proc. 22nd Int. Conf. Data Eng. (ICDE)*, Atlanta, GA, USA, 2006, p. 3.

[24] D. R. Musicant, J. M. Christensen, and J. F. Olson, "Supervised learning by training on aggregate outputs," in *Proc. 7th IEEE Int. Conf. Data Min. (ICDM)*, Omaha, NE, USA, 2007, pp. 252–261.

[25] N. Quadrianto, A. J. Smola, T. S. Caetano, and Q. V. Le, "Estimating labels from label proportions," *J. Mach. Learn. Res.*, vol. 10, pp. 2349–2374, Dec. 2009.

[26] S. Rüping, "SVM classifier estimation from group probabilities," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, Haifa, Israel, 2010, pp. 911–918.

[27] F. X. Yu, D. Liu, S. Kumar, T. Jebara, and S.-F. Chang, "$\propto$ SVM for learning with label proportions," in *Proc. 30th Int. Conf. Mach. Learn. (ICML)*, Atlanta, GA, USA, 2013, pp. 504–512.

[28] M. Stolpe and K. Morik, "Learning from label proportions by optimizing cluster model selection," in *Machine Learning and Knowledge Discovery in Databases*. Heidelberg, Germany: Springer, 2011, pp. 349–364.

[29] K. Fan *et al.*, "Learning a generative classifier from label proportions," *Neurocomputing*, vol. 139, pp. 47–55, Sep. 2014.

[30] G. Patrini, R. Nock, T. Caetano, and P. Rivera, "(Almost) no label no cry," in *Proc. Adv. Neural Inf. Process. Syst.*, Montréal, QC, Canada, 2014, pp. 190–198.

[31] F. X. Yu, K. Choromanski, S. Kumar, T. Jebara, and S.-F. Chang, "On learning from label proportions," *arXiv preprint arXiv:1402.5902*, 2014.

[32] J. Hernández-González, I. Inza, and J. A. Lozano, "Learning Bayesian network classifiers from label proportions," *Pattern Recognit.*, vol. 46, no. 12, pp. 3425–3440, 2013.

[33] S. Chen, B. Liu, M. Qian, and C. Zhang, "Kernel k-means based framework for aggregate outputs classification," in *Proc. IEEE Int. Conf. Data Min. Workshops (ICDMW)*, Miami, FL, USA, 2009, pp. 356–361.

[34] T. Cour, B. Sapp, and B. Taskar, "Learning from partial labels," *J. Mach. Learn. Res.*, vol. 12, pp. 1501–1536, May 2011.

[35] Z. Wang and J. Feng, "Multi-class learning from class proportions," *Neurocomputing*, vol. 119, pp. 273–280, Nov. 2013.

[36] T. Ni, F.-L. Chung, and S. Wang, "Support vector machine with manifold regularization and partially labeling privacy protection," *Inf. Sci.*, vol. 294, pp. 390–407, Feb. 2015.

[37] J. Hernández and I. Inza, "Learning naive Bayes models for multiple-instance learning with label proportions," in *Advances in Artificial Intelligence*. Heidelberg, Germany: Springer, 2011, pp. 134–144.

[38] T. Chen *et al.*, "Object-based visual sentiment concept analysis and application," in *Proc. ACM Int. Conf. Multimedia*, Orlando, FL, USA, 2014, pp. 367–376.

[39] F. X. Yu *et al.*, "Modeling attributes from category-attribute proportions," in *Proc. ACM Int. Conf. Multimedia*, Orlando, FL, USA, 2014, pp. 977–980.

[40] K.-T. Lai, F. X. Yu, M.-S. Chen, and S.-F. Chang, "Video event detection by inferring temporal instance labels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Columbus, OH, USA, 2014, pp. 2251–2258.

[41] P. S. Bradley and O. L. Mangasarian, "k-plane clustering," *J. Glob. Optim.*, vol. 16, no. 1, pp. 23–32, 2000.

[42] Jayadeva, R. Khemchandani, and S. Chandra, "Twin support vector machines for pattern classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 905–910, May 2007.

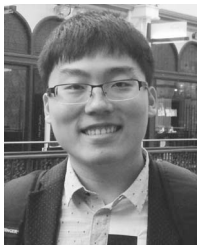
[43] Y. Tian, Z. Qi, X. Ju, Y. Shi, and X. Liu, "Nonparallel support vector machines for pattern classification," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1067–1079, Jul. 2014.
[44] S. Moro, R. Laureano, and P. Cortez, "Using data mining for bank direct marketing: An application of the crisp-DM methodology," in *Proc. Eur. Simulat. Model. Conf. (ESM)*, Guimarães, Portugal, 2011, pp. 117–121.

Zhiquan Qi received the master's and Ph.D. degrees from the College of Science, China Agricultural University, Beijing, China, in 2006 and 2011, respectively.

He is currently a Research Assistant with the Research Center on Fictitious Economy and Data Science, Chinese Academy of Sciences, Beijing. His current research interests include data mining, and the application in weak label learning.

Bo Wang received the master's degree from the Beijing Institute of Technology, Beijing, China, in 2010, and the Ph.D. degree from the University of Chinese Academy of Sciences, Beijing, in 2014.

He is currently with the School of Information and Technology and Management, University of International Business and Economics, Beijing. His current research interests include data mining, machine learning, and other data science related areas.



Fan Meng received the bachelor's degree from the Department of Information Management and Information System, Peking University, Beijing, China, in 2012. He is currently pursuing the Ph.D. degree with the University of Chinese Academy of Sciences, Beijing.

His current research interests include data mining, weak label learning, and its applications in computer vision.

Lingfeng Niu received the B.S. degree in mathematics from Xi'an Jiaotong University, Xi'an, China, in 2004, and the Ph.D. degree in mathematics from the Chinese Academy of Sciences, Beijing, China, in 2009.

She is currently an Associate Professor with the Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences. Her current research interests include optimization, machine learning, and data mining.