Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 18(6): 7727–7742.
DOI: 10.3934/mbe.2021383
Received: 04 June 2021
Accepted: 02 September 2021
Published: 07 September 2021

Research article

ℓ_1-norm based safe semi-supervised learning

Haitao Gan[1,2,*], **Zhi Yang**[1,3], **Ji Wang**[1] **and Bing Li**[4,*]

[1] School of Computer Science, Hubei University of Technology, Wuhan 430068, China

[2] Key Laboratory of Brain Machine Collaborative Intelligence of Zhejiang Province, Hangzhou 310018, China

[3] State Key Laboratory of Biocatalysis and Enzyme Engineering, Wuhan 430062, China

[4] School of Traffic and Transportation Engineering, Wuhan Institute of Shipbuilding Technology, Wuhan 430050, China

* **Correspondence:** Email: htgan01@qq.com, 278588198@qq.com.

Abstract: In the past few years, Safe Semi-Supervised Learning (S3L) has received considerable attentions in machine learning field. Different researchers have proposed many S3L methods for safe exploitation of risky unlabeled samples which result in performance degradation of Semi-Supervised Learning (SSL). Nevertheless, there exist some shortcomings: (1) Risk degrees of the unlabeled samples are in advance defined by analyzing prediction differences between Supervised Learning (SL) and SSL; (2) Negative impacts of labeled samples on learning performance are not investigated. Therefore, it is essential to design a novel method to adaptively estimate importance and risk of both unlabeled and labeled samples. For this purpose, we present ℓ_1-norm based S3L which can simultaneously reach the safe exploitation of the labeled and unlabeled samples in this paper. In order to solve the proposed ptimization problem, we utilize an effective iterative approach. In each iteration, one can adaptively estimate the weights of both labeled and unlabeled samples. The weights can reflect the importance or risk of the labeled and unlabeled samples. Hence, the negative effects of the labeled and unlabeled samples are expected to be reduced. Experimental performance on different datasets verifies that the proposed S3L method can obtain comparable performance with the existing SL, SSL and S3L methods and achieve the expected goal.

Keywords: semi-supervised learning; safe semi-supervised learning; performance degradation; ℓ_1 norm; importance estimation

1. Introduction

Along with the rapid development of Artificial Intelligence (AI) during the past few years, Semi-Supervised Learning (SSL) has attracted widespread attentions in machine learning field [1–6]. Various SSL methods are proposed to improve promising performance by exploiting both labeled and unlabeled samples. Different from Supervised Learning (SL), SSL tries to employ different assumptions or strategies to explore the unlabeled samples. The following assumptions are often used: (1) smoothness; (2) cluster; (3) manifold; (4)disagreement. SSL has shown the superiority to SL in some practical applications, such as object detection [7, 8], image classification [9–11], speech recognition [12, 13], etc. Although the SSL methods can achieve promising performance, they failed to consider the harmful effect of the unlabeled samples on the classification performance. Some previous studies [3, 14–18] have verified that the risky unlabeled samples can result in a bad consequence through theoretic and empirical analysis. The unlabeled samples may be risky which means that SSL is inferior to SL. If the risky unlabeled samples can not be safely explored, it will limit the application scope of SSL to some extent. Therefore, it is worthy to design Safe SSL (S3L) which never performs worse than SL.

Up to now, some researchers have proposed a few S3L methods which consider the risk of the unlabeled samples. On the whole, different approaches are proposed to conservatively explore the risky unlabeled samples. In our opinion, the proposed approaches can be summarized as follows: (1) Risky unlabeled samples are firstly identified through SSL and SL and then conservatively explored by SSL. (2) The risk degrees are firstly computed and then embedded into some SSL methods. (3) Multiple SSL classifiers are simultaneously learnt to decrease the degeneration probability.

In the first strategy, some methods tried to identify the risky unlabeled samples through SSL and SL. If an unlabeled sample was risky, it should be classified by SL. Otherwise, it would be classified by SSL. Thus the final classifier was trained by the labeled samples and unlabeled ones with the pseudo labels predicted by SL or SSL. In particular, Li and Zhou [19] presented an S3VM_*us* method where a hierarchical clustering method was used to select the risky unlabeled samples. The selected samples were then predicted by SVM and the remaining were predicted by transductive SVM (TSVM) [20]. Hence, S3VM_*us* should have a smaller degeneration probability than TSVM. Li et al. [21] built a large margin approach named LargE margin grAph quality juDgement (LEAD). LEAD firstly constructed multiple graphs and then performed Graph-based SSL (GSSL) to yield the predictions of the labeled and unlabeled samples. The risky unlabeled samples were identified by SVM which considered the predictions of multiple GSSL as the input features. Finally, their pseudo labels in LEAD were predicted by SVM and the rest were labeled according to the average predictions of multiple GSSL.

In the second strategy, some regularization approaches were used to exploit the risky unlabeled samples. Wang and Chen [22] introduced Safety-Aware SSCCM (SA-SSCCM) which was extended from semi-supervised classification method based on class membership (SSCCM). SA-SSCCM utilized a ℓ_2-norm based loss function to restrict the predictions of the unlabeled samples to be those of SL (i.e., Least-Square SVM (LS-SVM)). The performance of SA-SSCCM was never significantly inferior to that of LS-SVM and seldom significantly inferior to that of SSCCM. Gan et al. [15] proposed Risk-based Safe Laplacian Regularized Least Squares (RsLapRLS) which tried to assign risk degrees to different unlabeled samples and a risk-based regularization term was embedded into LapRLS to reduce the risk. Wang et al. [23] proposed safe LS_S3VM based on Adjusted Cluster

Assumption (ACA-S3VM) which discussed the negative effect of an inappropriate model assumption (e.g., cluster assumption). In this method, the unlabeled samples lied in the cluster boundary were found by unsupervised clustering and cautiously explored.

Different from the aforementioned two strategies which learned one single classifier, the third strategy tried to simultaneously learn multiple semi-supervised classifiers or regressors. Li and Zhou [24] developed safe semi-supervised SVMs (S4VMs). S4VMs constructed multiple low-density separators simultaneously to reduce the risk of identifying a poor separator with unlabeled samples. The performance of S4VMs was never significantly inferior to that of SVM. Furthermore, in order to extend S4VMs for dealing with multi-class problems, Thiago et al. [25] proposed a hierarchical bottom-up S4VMs tree scheme to take advantage of S4VMs. Similar to S4VMs, Gan et al. [26] presented a Safety-aware GSSL (SaGSSL) method which tried to learn a safe classifier by constructing multiple graphs. The experimental results demonstrated that SaGSSL could adaptively select good graphs from the candidates and reduce the risk of inappropriate graphs for GSSL. Except the classification problem, Li et al. [27] proposed SAFE semi-supervised Regression (SAFER) which was designed to deal with the regression problem. SAFER aimed to learn a safe regressor from multiple semi-supervised ones and obtained the desired performance.

Although many different S3L methods have been proposed to safely explore the unlabeled samples and obtained better performance compared to SL and SSL, they have the following shortcomings: (1) The risk degrees of the unlabeled samples are given by an pre-defined approach; (2) The hurt of the labeled samples (e.g., mislabeled ones) on learning performance is not considered.

To overcome the shortcomings, we present ℓ_1-norm based S3L which can simultaneously reach the safe exploitation of the labeled and unlabeled samples. In the proposed algorithm, we utilize a ℓ_1-norm based loss function to generate the objective function of S3L and use an effective iterative optimization technique to obtain an optimal solution. In each iteration, the weights of the labeled and unlabeled samples are adaptively estimated. The risky samples will have small weights and a small impact on training the final classifier. Hence, it is expected to alleviate the negative influence of both labeled and unlabeled samples. The main contributions of the proposed algorithm include: (1) The risk of both labeled and unlabeled samples can be reduced by adaptively the weights through ℓ_1 norm; (2) An effective optimization algorithm is introduced to solve the proposed problem. This work is an extended version of our work in [28]. In comparison to the work in [28], we have given more details of the existing related methods and proposed algorithm. Additionally, extensive experiments are performed on more datasets to show the effectiveness of the proposed algorithm, including result discussion and parameter analysis.

The remaining part of the paper is organized as follows: In Section 2, we firstly review background knowledge (i.e., LapRLS and RsLapRLS). In Section 3, we will present the motivation and give the details of proposed algorithm. Section 4 will report the results to verify the effectiveness of the proposed algorithm by conducting experiments on several UCI and benchmark datasets. Finally, we will present the conclusions and some future work in Section 5.

2. Background knowledge

2.1. Laplacian regularized least squares

Suppose a dataset $X = \{(x_1, y_1), \cdots, (x_l, y_l), x_{l+1}, \cdots, x_n\}$ with l labeled samples and $u = n - l$

unlabeled ones, $x_i \in \mathbb{R}^{\mathrm{D}}$ and $y_i \in \{-1, 1\}$. For the purposes of exploration of unlabeled samples, LapRLS [29] tried to construct a local graph W. The graph was used to approximate to geometric structure of the whole samples. The graph W was constructed as:

$$W_{ij} = \begin{cases} \exp\{-\frac{\|x_i - x_j\|_2^2}{2\sigma^2}\} & \text{if } x_i \in N_p(x_j) \text{ or } x_j \in N_p(x_i) \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

here $N_p(x_i)$ denotes the data sets of p nearest neighbors of x_i.

After obtaining the graph, LapRLS built a graph-based regularization term and embedded it into RLS. The term was written as:

$$\mathcal{R} = \frac{1}{2}\sum_{i=1}^{n}(f(x_i) - f(x_j))^2 W_{ij} = f^T L f \tag{2.2}$$

here $f = [f(x_1), \cdots, f(x_n)]^T$, and L is the graph Laplacian defined as $L = D - W$ with $D_{ii} = \sum_j W_{ij}$.

The objective function of LapRLS was then written as:

$$\mathcal{J}(f) = \sum_{i=1}^{l}(f(x_i) - y_i)^2 + \gamma_A \|f\|_K^2 + \gamma_I f^T L f \tag{2.3}$$

where $\|\cdot\|_K$ is the norm defined in $\mathcal{H}_K$ which is a Reproducing Kernel Hilbert Space (RKHS) associated with a Mercer kernel $K : X \times X \to \mathbb{R}$. γ_A and γ_I are regularization parameters. When γ_I is equal to 0, LapRLS will boil down to RLS.

According to the Representer Theorem [29], the decision function was represented as

$$f(x) = \sum_{i=1}^{n} \alpha_i^* k(x_i, x) \tag{2.4}$$

where α^* denoted the optimal value of α and $k(\cdot, \cdot)$ was the Mercer kernel.

The equation Eq (2.3) was rewritten as:

$$\mathcal{J}(\alpha) = (K_l\alpha - Y)^T(K_l\alpha - Y) + \gamma_A \alpha^T K \alpha + \gamma_I \alpha^T K L K \alpha \tag{2.5}$$

where $Y = [y_1, \cdots, y_l]^T$ was the given labels. The Gram matrix K was calculated using a Mercer kernel whose entry $K_{ij} = k(x_i, x_j)$. K_l was the first l rows and k_u was the last u rows in K.

By setting the derivative of Eq (2.5) to zero, we can obtain the following solution:

$$\alpha^* = (K_l^T K_l + \gamma_A K + \gamma_I K L K)^{-1}(K_l^T Y) \tag{2.6}$$

2.2. Risk-based safe laplacian regularized least squares

In order to reach the safe exploitation of the unlabeled samples, RsLapRLS utilized a ℓ_2 norm to define the loss function. The loss function constrained the outputs of the unlabeled samples to be those of RLS. Hereafter, we denote the supervised classifier obtained by RLS as $g(x)$. The objective function of RsLapRLS was given as:

$$\begin{aligned} Q(f) = &\sum_{i=1}^{l}(f(x_i) - y_i)^2 + \gamma_A \|f\|_K^2 + \gamma_I f^T L f \\ &+ \lambda \sum_{j=l+1}^{n} s_j (f(x_j) - g(x_j))^2 \end{aligned} \tag{2.7}$$

where λ was a regularization parameter. Specifically, RsLapRLS became RLS when γ_I and λ were both 0 and became LapRLS when $\lambda = 0$. s_j was the risk degree of the unlabeled sample x_j.

From the above equation, the output of RsLapRLS was a balance between that of RLS and LapRLS. A risk degree estimation equation was pre-defined for the unlabeled sample x_j.

$$s_j = \begin{cases} \exp\{-cf(x_j)\} & \text{if } cs(x_j) = 1 \text{ and } cf(x_j) \neq 0 \\ \exp\{|cf(x_j)|\} & \text{if } cs(x_j) = -1 \text{ and } cf(x_j) \neq 0 \\ \exp\{-cs(x_j)\} & \text{if } cf(x_j) = 0 \end{cases} \tag{2.8}$$

here, the *consistency* ($cs(x_j)$) and *confidence* ($cf(x_j)$) were calculated as:

$$\begin{aligned} cs(x_j) &= sign(\widehat{y}_j \cdot \overline{y}_j) \\ cf(x_j) &= |\widehat{y}_j| - |\overline{y}_j| \end{aligned} \tag{2.9}$$

where $\widehat{y}_j$ and $\overline{y}_j$ were respectively the outputs of LapRLS and RLS.

$cs(x_j)$ denoted the prediction consistency of x_j between LapRLS and RLS. If the predictions were consistent, $cs(x_j)$ was equal to 1, otherwise -1. $cf(x_j)$ represented the prediction confidence or difference of x_j between LapRLS and RLS.

As shown in Eqs (2.8) and (2.9), the risk degree s_j was small when $cs(x_j) = 1$ and $cf(x_j) > 0$ or $cs(x_j) = 1$ and $cf(x_j) = 0$. In this case, the unlabeled sample x_j should be exploited through the semi-supervised approach. Otherwise, the risk degree s_j was large and the output of x_j should approach to that of RLS which was defined by the last term in Eq (2.7).

According to Eq (2.4), the objective function Eq (2.7) was then rewritten as:

$$\begin{aligned} Q(\alpha) =&(K_l\alpha - Y)^T(K_l\alpha - Y) + \gamma_A\alpha^T K\alpha + \gamma_I\alpha^T KLK\alpha \\ &+ \lambda(K_u\alpha - \overline{Y})^T S(K_u\alpha - \overline{Y}) \end{aligned} \tag{2.10}$$

where $\overline{Y} = [g(x_{l+1}), \cdots, g(x_n)]^T$ was the outputs of RLS and S was a diagonal matrix with $S_{jj} = s_{j+l}$.

The derivative of Eq (2.10) with respect to α could be otained and set to zero. The solution of RsLapRLS was then given as:

$$\begin{aligned} \alpha^* =&(K_l^T K_l + \gamma_A K + \gamma_I KLK + \lambda K_u^T S K_u)^{-1} \\ &(K_l^T Y + \lambda K_u^T S\overline{Y}) \end{aligned} \tag{2.11}$$

3. Proposed algorithm

3.1. Motivation

As shown in Section 1, the existing S3L methods mainly consider the hurt of the unlabeled samples. Firstly, the risk degrees of the unlabeled samples are in advance defined in S3L. If inappropriate risk degrees are assigned to the unlabeled samples, the unlabeled samples will not be safely exploited. Therefore, it is important to investigate an adaptive weight computing method. Secondly, the risk of the labeled samples should be considered. In some cases, the samples may be mislabeled by the experts. In the next subsection, we will present how to solve the above problems using ℓ_1 norm.

3.2. Formulation and solution

For convenience, the objective function of RsLapRLS is written as:

$$Q(f) = ||f_l - Y||_2^2 + \gamma_A ||f||_K^2 + \gamma_I f^T L f + \lambda ||f_u \circ s^{\frac{1}{2}} - \overline{Y} \circ s^{\frac{1}{2}}||_2^2 \tag{3.1}$$

where f_l and f_u respectively denote the outputs of the labeled and unlabeled samples. $s = [s_{l+1}, \cdots, s_n]$ denotes the risk degrees of the unlabeled samples. $A \circ B$ represents the Hadamard product between the vectors A and B.

In this paper, a ℓ_1 norm is used to substitute for the ℓ_2 norm. Hence, the objective function of the proposed algorithm is written as:

$$\mathcal{J}(f) = ||f_l - Y||_1 + \gamma_A ||f||_K^2 + \gamma_I f^T L f + \lambda ||f_u - \overline{Y}||_1 \tag{3.2}$$

Equation (3.2) can be written as:

$$\begin{aligned}\mathcal{J}(f) &= \sum_{i=1}^{l} |f(x_i) - y_i| + \gamma_A ||f||_K^2 + \gamma_I f^T L f \\ &+ \lambda \sum_{j=l+1}^{n} |f(x_j) - g(x_j)| \\ &= \sum_{i=1}^{n} r_i |f(x_i) - \tilde{y}_i| + \gamma_A ||f||_K^2 + \gamma_I f^T L f \\ &= \sum_{i=1}^{n} r_i \sqrt{(f(x_i) - \tilde{y}_i)^2} + O(f)\end{aligned} \tag{3.3}$$

where $\tilde{y}_i = y_i$ and $r_i = 1$ if $i \in \{1, \cdots, l\}$ and $\tilde{y}_i = g(x_i)$ and $r_i = \lambda$ if $i \in \{l+1, \cdots, n\}$.

The derivative of Eq (3.3) with respect to $f(x_i)$ is obtained as:

$$\frac{\partial \mathcal{J}(f)}{\partial f(x_i)} = \sum_{i=1}^{n} \mu_i r_i \frac{\partial (f(x_i) - \tilde{y}_i)^2}{\partial f(x_i)} + \frac{\partial O(f)}{\partial f(x_i)} \tag{3.4}$$

where $\mu_i = \frac{1}{2|f(x_i) - \tilde{y}_i| + \varepsilon}$ with a small value ε.

If μ_i is fixed, Eq (3.4) can be considered as the derivative with respect to $f(x_i)$ of the following function:

$$\mathcal{R}(f) = \sum_{i=1}^{n} \mu_i r_i (f(x_i) - \tilde{y}_i)^2 + O(f) \tag{3.5}$$

As we know, a closed-form solution of Eq (3.5) can be obtained. Since μ_i is related to $f(x_i)$, an alternating optimization approach is utilized to solve the optimization problem (3.3). In each iteration, f and μ_i are calculated, respectively. Because the computation formula of μ_i has been given, we next discuss how to calculate f when μ_i is known.

Based on Eq (2.4), Eq (3.5) can be denoted as:

$$\mathcal{R}(\alpha) = (K\alpha - \widetilde{Y})^T R (K\alpha - \widetilde{Y}) + \gamma_A \alpha^T K \alpha + \gamma_I \alpha^T K L K \alpha \tag{3.6}$$

where $\widetilde{Y} = [\tilde{y}_1, \cdots, \tilde{y}_n]^T$ and R is a diagonal matrix with $R_{ii} = \mu_i r_i$.

The derivative of Eq (3.6) with respect to α is obtained and set to zero, the following equation is achieved:

$$\alpha^* = (RK + \gamma_A I + \gamma_I LK)^{-1}(R\widetilde{Y}) \tag{3.7}$$

It is worthy to point out that the weight μ_i can reflect the risk degree of the labeled and unlabeled samples. If the difference between the prediction and given label for a labeled sample is large, the labeled sample may be mislabeled and risky. In this case, the corresponding μ_i should be small. Meanwhile, the unlabeled samples may be safe if the corresponding output approaches to that of LS and the weight μ_i is large. Otherwise, the unlabeled sample should be risky and the weight μ_i will be small. Hence the risky labeled and unlabeled samples will have a small impact on training the final classifier. Overall, it is expect to yield a safe exploitation through the weight.

The iterative framework of the proposed algorithm is shown in Algorithm 1.

Algorithm 1 Proposed algorithm

Input: A dataset $X = \{(x_1, y_1), \cdots, (x_l, y_l), x_{l+1}, \cdots, x_n\}$, the parameters γ_A, γ_I, λ, η, and $Maxiter$.
Output: The optimal decision coefficients α^*.

```
Train a RLS classifier $g(x)$ and calculate the outputs of the unlabeled samples using $g(x)$;
Initialize $\mu_i^{(0)} = 1, i = 1, \cdots, n$;
for $t = 1 : Maxiter$ do
    Update $\alpha^{(t)}$ using Eq (3.7);
    Compute $f^t(x_i) = \sum_{j=1}^n \alpha_j^{(t)} k(x_i, x_j)$;
    Update $\mu_i^{(t)} = \frac{1}{2|f^t(x_i) - \tilde{y}_i| + \varepsilon}$;
    if $\|\mu^{(t)} - \mu^{(t-1)}\| < \eta$ then
        return $\alpha^{(t)}$.
    end if
end for
```

3.3. The convergence analysis

Next, we will give a convergence analysis of Algorithm 1. Eq (3.5) can be written as

$$\mathcal{R}(\alpha) = \sum_{i=1}^{n} \mu_i r_i (K_i \alpha - \tilde{y}_i)^2 + \gamma_A \alpha^T K \alpha + \gamma_I \alpha^T KLK\alpha \tag{3.8}$$

here K_i is the i-th row of the matrix K.

First, the optimal solution α of the optimization problem Eqs (3.8) and (3.6) can be yield through Eq (3.7). Hence, in the t-th iteration, $\alpha^{(t)}$ is the optimal solution of Eq (3.8). We have

$$\begin{aligned} &\sum_{i=1}^{n} \mu_i^{(t-1)} r_i (K_i \alpha^{(t)} - \tilde{y}_i)^2 + \gamma_A (\alpha^{(t)})^T K \alpha^{(t)} + \gamma_I (\alpha^{(t)})^T KLK\alpha^{(t)} \\ \leq &\sum_{i=1}^{n} \mu_i^{(t-1)} r_i (K_i \alpha^{(t-1)} - \tilde{y}_i)^2 + \gamma_A (\alpha^{(t-1)})^T K \alpha^{(t-1)} + \gamma_I (\alpha^{(t-1)})^T KLK\alpha^{(t-1)} \end{aligned} \tag{3.9}$$

Since $\mu_i = \frac{1}{2|f(x_i)-\tilde{y}_i|}$, the above inequality can be written as

$$\begin{aligned}&\sum_{i=1}^{n} \frac{r_i(K_i\alpha^{(t)} - \tilde{y}_i)^2}{2|K_i\alpha^{(t-1)} - \tilde{y}_i|} + \gamma_A(\alpha^{(t)})^T K\alpha^{(t)} + \gamma_I(\alpha^{(t)})^T KLK\alpha^{(t)} \\ \leq &\sum_{i=1}^{n} \frac{r_i(K_i\alpha^{(t-1)} - \tilde{y}_i)^2}{2|K_i\alpha^{(t-1)} - \tilde{y}_i|} + \gamma_A(\alpha^{(t-1)})^T K\alpha^{(t-1)} + \gamma_I(\alpha^{(t-1)})^T KLK\alpha^{(t-1)}\end{aligned} \tag{3.10}$$

Because

$$|K_i\alpha^{(t)} - \tilde{y}_i| - \frac{(K_i\alpha^{(t)} - \tilde{y}_i)^2}{2|K_i\alpha^{(t-1)} - \tilde{y}_i|} \leq |K_i\alpha^{(t-1)} - \tilde{y}_i| - \frac{(K_i\alpha^{(t-1)} - \tilde{y}_i)^2}{2|K_i\alpha^{(t-1)} - \tilde{y}_i|} \tag{3.11}$$

and r_i is a nonnegative constant, we have

$$\sum_{i=1}^{n}\left(r_i|K_i\alpha^{(t)} - \tilde{y}_i| - \frac{r_i(K_i\alpha^{(t)} - \tilde{y}_i)^2}{2|K_i\alpha^{(t-1)} - \tilde{y}_i|}\right) \leq \sum_{i=1}^{n}\left(r_i|K_i\alpha^{(t-1)} - \tilde{y}_i| - \frac{r_i(K_i\alpha^{(t-1)} - \tilde{y}_i)^2}{2|K_i\alpha^{(t-1)} - \tilde{y}_i|}\right) \tag{3.12}$$

Through Eqs (3.10) and (3.12), the following inequality is achieved.

$$\begin{aligned}&\sum_{i=1}^{n} r_i|K_i\alpha^{(t)} - \tilde{y}_i| + \gamma_A(\alpha^{(t)})^T K\alpha^{(t)} + \gamma_I(\alpha^{(t)})^T KLK\alpha^{(t)} \\ \leq &\sum_{i=1}^{n} r_i|K_i\alpha^{(t-1)} - \tilde{y}_i| + \gamma_A(\alpha^{(t-1)})^T K\alpha^{(t-1)} + \gamma_I(\alpha^{(t-1)})^T KLK\alpha^{(t-1)}\end{aligned} \tag{3.13}$$

From Eqs (3.2), (3.3) and (3.13), one can see that $\mathcal{J}(f)$ will be monotonically decreased and nonnegative. Therefore, Algorithm 1 will be converged.

4. Experimental analysis

Next, we perform some experiments to show the behavior of the proposed algorithm. The used datasets are selected from UCI [*] and benchmark ones [1]. Firstly, two subsets are obtained by randomly dividing the whole dataset. The two subsets include 10 or 100 labeled samples and the remaining unlabeled ones. This division process is repeated 30 times for the UCI datasets and 12 times for the benchmark datasets. The details of each dataset are described in Table 1. In the experiments, the following methods are used:

- SL: RLS, SVM, LS-SVM.
- SSL: LapRLS, TSVM, SSCCM.
- S3L: S3VM_*us*, S4VMs, SA-SSCCM, RsLapRLS.

For the UCI datasets, the parameters C_1 and C_2 in S4VMs are respectively fixed to 1 and 0.1. λ_1, λ_2 and η in SA-SSCCM are set to 100, 1 and 1, respectively. For benchmark datasets, C_1 and C_2 in S4VMs are respectively set to 100 and 0.1. λ_1, λ_2 and η in SA-SSCCM are set to 100, 0.1 and 1, respectively. ϵ in S3VM_*us* is set to 0.1. γl in RLS is fixed to 0.05, p in LapRLS, RsLapRLS and

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

Table 1. Description of the datasets.

ID	Dataset	#Points	#Features
1	Heart	270	13
2	Diabetes	768	8
3	German	1000	20
4	BCI	400	114
5	G241c	1500	241
6	G241n	1500	241
7	Digit1	1500	241
8	USPS	1500	241

proposed algorithm is set to 5. γ_A, γ_I and λ in LapRLS, RsLapRLS and proposed algorithm are obtained by leave-one-out cross validation in the case of 10 labeled samples and by 5-fold cross validation in the case of 100 labeled samples. γ_A and γ_I are selected in the set $\{10^{-6}, 10^{-4}, 10^{-2}, 1, 10, 100\}$ and λ is selected from $\{10^{-6}, 10^{-4}, 10^{-2}, 0.1, 0.5, 0.9, 1, 5, 9\}$. The linear and Gaussian kernels are used to calculate the Gram matrix K. In the Gaussian kernel, the width parameter δ is set to the average distance between the samples in the case of 10 labeled samples, and obtained by 5-fold cross validation among $\{0.25\delta, 0.5\delta, \delta, 2\delta, 4\delta\}$ in the case of 100 labeled samples.

Tables 2 and 3 report the classification results of different methods. The third-to-last row reports the average accuracies of different methods. The second-to-last row reports the win/tie/loss (W/T/L) counts where SSL performs better/comparable/worse than the corresponding SL. The last row reports the W/T/L counts where S3L performs better/comparable/worse than the corresponding SSL.

As can be seen from the two tables, it can be concluded:

(a) According to the average accuracy, the proposed algorithm can perform better than the other SL and SSL methods. On the one hand, it demonstrates that the unlabeled samples can generally boost the learning performance. On the other hand, the proposed algorithm can be used to learn a semi-supervised classifier.

(b) In terms of the W/T/L counts, different SSL methods perform worse than the corresponding SL methods in some cases. textcolor[rgb]1,0,0Specifically, LapRLS performs worse than RLS on 18 out of 32 cases. It shows the negative influence of the unlabeled samples on the learning performance.

(c) In terms of the W/T/L counts, the S3L methods are never significantly inferior to the corresponding SL methods. In particular, RsLapRLS and the proposed algorithm can perform better or comparable to RLS on all cases. It is indicated that the proposed algorithm can reduce the risk of the unlabeled samples.

(d) Finally, the proposed algorithm is superior and more robust than RsLapRLS on all cases in term of the average accuracy and standard deviation. We can conclude that the proposed ℓ_1-norm strategy is effective to deal with the hurt of the labeled samples. Since the weights of both labeled and unlabeled samples are adaptively estimated, it will further alleviate the negative impact which is caused by artificial risk degree assignment in RsLapRLS.

Table 2. Classification performance with 10 labeled samples.

Linear	RLS	SVM	LS-SVM	LapRLS	TSVM	SSCCM	S3VM_*us*	S4VMs	SA-SSCCM	RsLapRLS	Proposed
1	69.1±8.2	63.6±8.0	65.7±7.3	76.1±3.9	63.9±7.5	68.2±11.6	63.8±8.2	63.9±7.8	68.0±10.2	79.4±9.6	80.9±3.4
2	64.7±6.9	58.8±2.8	59.4±6.9	68.7±6.0	57.7±1.0	58.2±2.6	58.8±2.8	58.2±3.4	59.6±2.6	71.6±7.2	72.3±4.9
3	55.2±8.2	60.1±6.0	61.5±4.8	55.7±6.8	56.2±9.2	62.6±6.9	61.6±6.2	60.2±5.6	62.2±6.8	55.9±10.3	64.8±6.3
4	53.7±3.5	52.8±1.9	52.6±3.1	53.4±2.5	51.3±3.4	53.6±2.6	52.6±2.3	52.0±2.3	54.8±3.0	57.2±2.9	56.8±2.7
5	61.4±3.4	54.6±4.4	61.3±3.5	55.7±2.0	78.4±5.8	63.2±3.4	54.7±2.9	54.6±4.2	63.8±3.6	64.0±4.0	64.2±4.1
6	59.6±4.1	56.4±5.2	59.5±4.1	57.1±2.4	53.4±9.6	55.4±7.4	56.4±5.4	56.3±5.1	59.7±4.4	60.7±5.1	60.6±4.8
7	76.9±5.2	77.0±5.2	65.4±10.8	65.4±8.8	79.7±3.3	65.6±9.1	77.4±5.4	76.9±5.1	66.5±8.4	79.3±6.2	79.1±5.9
8	77.7±3.1	77.9±3.0	79.9±0.9	81.0±1.3	72.4±3.7	80.1±0.4	78.0±2.7	78.2±2.9	80.3±0.7	80.3±1.5	81.9±1.4
Average	64.8	62.7	63.2	64.1	64.1	63.4	62.9	62.5	64.4	68.6	70.1
Semi-supervised vs. cor. supervised: W/T/L				3/2/3	2/1/5	4/2/2	1/7/0	0/8/0	4/4/0	7/1/0	8/0/0
Safe semi-supervised vs. cor. semi-supervised: W/T/L							5/1/2	3/3/2	3/5/0	6/2/0	8/0/0
Gaussian	RLS	SVM	LS-SVM	LapRLS	TSVM	SSCCM	S3VM_*us*	S4VMs	SA-SSCCM	RsLapRLS	Proposed
1	73.9±5.9	61.9±3.9	61.8±5.1	68.2±5.2	61.4±4.8	64.1±4.9	62.1±3.9	62.2±4.1	64.1±4.9	76.6±6.3	80.1±3.5
2	67.0±6.1	59.1±3.1	59.8±5.2	64.0±5.2	51.7±1.0	59.6±2.6	58.8±2.8	57.6±2.6	59.7±2.6	70.3±4.0	72.3±2.1
3	56.2±7.2	60.6±5.6	59.6±8.8	51.0±9.6	53.3±6.3	62.8±6.9	61.1±5.1	60.8±5.4	62.5±6.9	64.8±10.7	67.6±5.6
4	52.9±2.2	51.1±2.9	50.9±0.8	50.5±1.5	51.4±2.7	53.0±2.4	51.1±2.7	51.2±2.7	53.0±2.4	54.1±1.7	54.7±1.7
5	53.5±4.5	53.6±4.4	52.4±4.6	49.9±0.3	59.0±4.8	49.9±0.1	53.6±4.6	53.8±4.4	57.5±2.8	54.6±6.5	55.3±6.1
6	55.8±5.3	52.8±5.2	52.2±4.6	50.1±0.4	53.1±6.7	49.9±0.2	52.9±5.0	53.0±5.4	58.3±3.0	58.4±6.2	56.6±5.9
7	76.4±6.8	57.5±6.5	56.1±12.1	78.1±17.8	80.3±3.1	66.3±10.7	60.2±6.7	76.8±6.4	66.6±10.6	83.1±9.8	87.8±10.6
8	79.8±2.3	79.9±2.2	80.0±0.1	80.1±0.3	71.3±2.6	81.1±0.9	79.5±1.9	79.3±2.3	81.4±1.1	81.6±1.9	82.0±1.7
Average	64.4	59.6	59.1	61.5	60.2	60.8	59.9	61.8	62.9	67.9	69.6
Semi-supervised vs. cor. supervised: W/T/L				1/1/6	2/3/3	5/1/2	2/6/0	1/6/1	7/1/0	8/0/0	8/0/0
Safe semi-supervised vs. cor. semi-supervised: W/T/L							3/3/2	3/3/2	2/6/0	8/0/0	8/0/0

Table 3. Classification performance with 100 labeled samples.

Linear	RLS	SVM	LS-SVM	LapRLS	TSVM	SSCCM	S3VM_*us*	S4VMs	SA-SSCCM	RsLapRLS	Proposed
1	80.8±2.2	82.3±2.2	83.8±1.5	81.1±2.5	81.9±1.3	84.1±1.5	82.9±2.1	82.0±1.8	84.7±1.6	83.3±2.2	84.0±2.1
2	74.6±2.2	73.4±2.2	73.8±1.3	74.2±1.8	70.6±6.9	76.1±1.6	73.6±2.1	74.8±1.8	77.1±0.9	75.9±2.4	75.8±1.7
3	66.6±2.4	69.9±2.2	63.5±1.9	65.9±2.4	58.2±18.7	72.2±0.1	69.5±1.7	70.1±1.5	70.3±0.5	72.3±2.6	72.7±1.7
4	71.3±3.3	71.8±3.5	73.1±2.9	64.2±2.1	71.5±4.1	75.3±2.4	71.6±3.2	71.5±3.4	76.8±2.9	70.8±3.1	75.6±2.1
5	72.5±2.2	75.0±1.9	76.3±1.5	67.1±1.7	80.0±1.7	73.9±2.2	75.0±1.9	75.2±2.5	75.6±1.8	75.5±2.0	76.3±1.5
6	70.7±2.4	72.4±2.8	73.1±3.1	67.5±1.9	76.3±2.4	73.8±2.7	72.8±1.8	74.9±2.3	72.9±2.1	73.2±2.5	74.3±1.6
7	85.6±1.7	92.4±1.4	92.3±1.5	90.6±1.3	92.5±1.9	91.6±1.8	92.4±1.4	92.6±2.1	92.5±1.5	92.8±2.1	92.2±1.4
8	84.4±0.9	88.2±0.9	88.0±0.8	82.2±1.2	86.7±1.4	86.6±1.1	88.1±0.9	88.5±1.3	87.8±1.1	88.5±1.2	88.0±0.9
Average	75.8	78.2	78.0	74.1	77.2	79.2	78.2	78.7	79.7	79.0	79.9
Semi-supervised vs. cor. supervised: W/T/L				1/3/4	2/3/3	3/3/2	0/8/0	2/6/0	3/5/0	7/1/0	8/0/0
Safe semi-supervised vs. cor. semi-supervised: W/T/L							3/3/2	3/3/2	3/4/1	8/0/0	8/0/0
Gaussian	RLS	SVM	LS-SVM	LapRLS	TSVM	SSCCM	S3VM_*us*	S4VMs	SA-SSCCM	RsLapRLS	Proposed
1	80.0±4.2	79.6±0	84.7±0	78.5±3.0	80.0±0	86.5±0	79.4±0	80.5±0	86.9±0	83.6±2.2	93.9±2.1
2	72.5±3.1	73.2±1.8	75.1±1.0	69.7±1.9	71.2±1.3	74.2±1.0	72.6±1.0	73.4±1.0	75.2±1.9	74.7±2.0	74.3±1.8
3	65.9±3.2	71.6±3.4	67.6±2.9	60.5±2.6	53.1±3.2	72.7±3.7	71.0±2.3	71.4±2.9	71.8±3.1	73.5±1.6	73.8±1.5
4	67.2±4.1	67.4±3.2	68.6±2.3	52.2±3.0	65.8±2.8	70.2±3.1	67.2±3.4	67.2±2.8	69.8±3.8	68.7±4.7	67.2±4.8
5	76.6±2.1	69.6±6.8	68.4±5.1	53.4±3.9	78.1±2.2	77.3±4.1	69.8±3.2	75.2±2.9	75.2±3.3	80.2±3.6	79.6±3.3
6	74.7±2.0	61.3±8.6	64.2±7.8	56.6±4.0	66.3±5.8	62.8±6.9	61.9±6.2	62.3±3.3	64.1±3.6	76.9±3.0	76.8±2.8
7	94.6±1.6	95.1±1.6	94.8±1.8	97.9±0.5	95.5±1.5	95.5±1.6	95.1±1.4	96.2±1.3	95.8±1.8	98.1±0.6	98.1±0.5
8	90.7±2.0	84.6±2.1	85.7±1.8	90.5±2.6	90.8±1.5	87.9±2.6	86.8±2.8	89.4±1.5	88.2±2.1	92.5±2.1	92.5±1.9
Average	77.8	75.3	76.1	69.9	75.1	78.4	75.5	77.0	78.4	81.0	82.0
Semi-supervised vs. cor. supervised: W/T/L				1/2/5	3/2/3	5/2/1	1/7/0	3/5/0	5/3/0	8/0/0	7/1/0
Safe semi-supervised vs. cor. semi-supervised: W/T/L							3/2/3	3/2/3	1/6/1	7/1/0	7/1/0

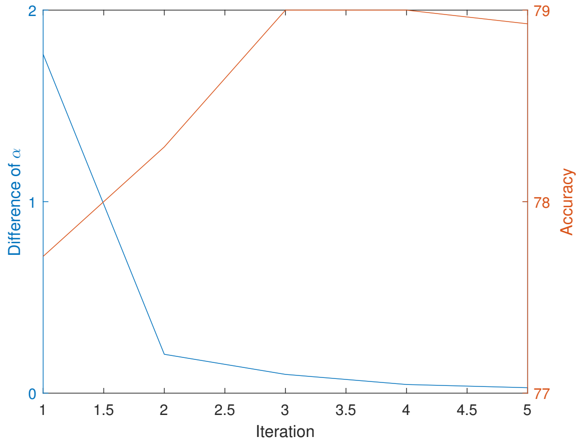

Figure 1. A plot of convergence and testing accuracy on G241n dataset.

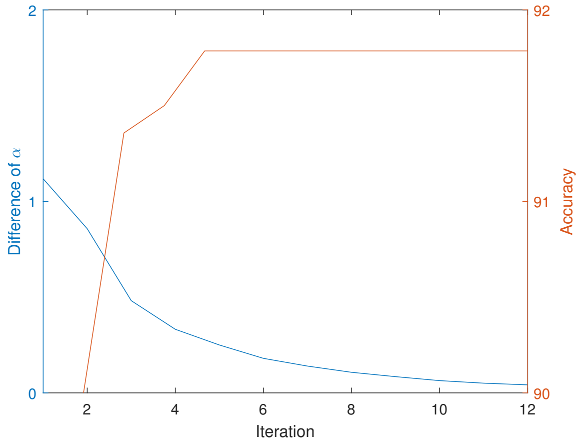

Figure 2. A plot of convergence and testing accuracy on USPS dataset.

Additionally, we carry out some experiments to analyze the convergence of the proposed algorithm. Figures 1 and 2 give the plots of convergence and the corresponding testing accuracy at each iteration on G241n and USPS datasets. From the two figures, one can see that the porposed algorithm usually converges after less than 15 iterations. It shows that the proposed algorithm is efficient and converges fast.

Futhermore, we give a parameter analysis which tries to analyze the impact of λ on the performance of the proposed algorithm. The parameter λ can reflect the importance of the regularizer constraining the outputs of the risky unlabeled samples. The value is selected among $\{10^{-6}, 10^{-4}, 10^{-2}, 0.1, 0.5, 0.9, 1, 5, 9\}$. Figures 3–6 report the accuracies of the proposed algorithm as the parameter λ changes. From these figures, one can see that proposed algorithm can obtain the best results in a wide range. It will extend the practicability of proposed algorithm to some extent.

5. Conclusions

On the whole, we have investigated a ℓ_1-norm based S3L algorithm. This algorithm has ultilized ℓ_1 norm instead of ℓ_2 norm to define the objective function. It can effectively reduce the risk of both

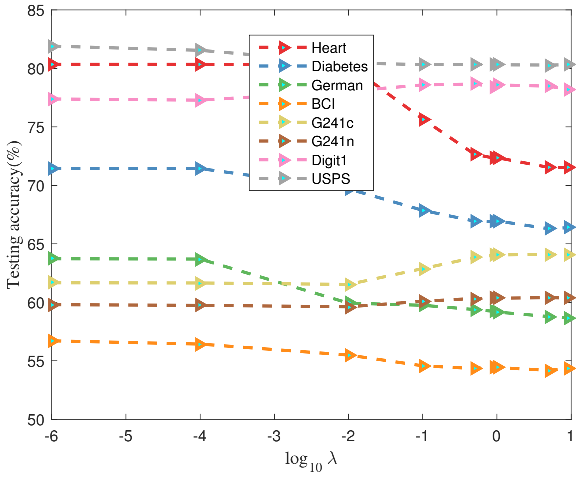

Figure 3. Accuracy for 10 labeled data with linear kernel.

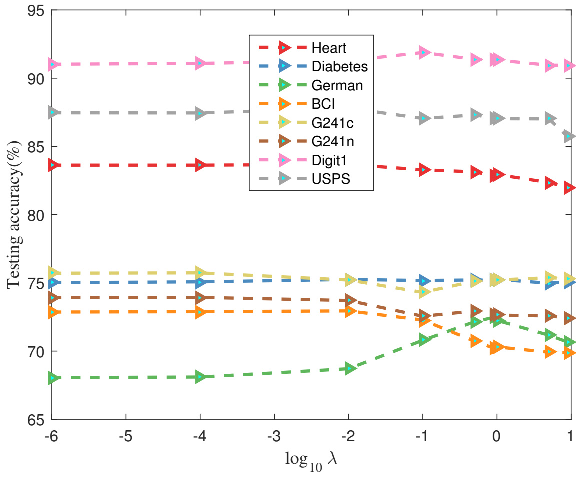

Figure 4. Accuracy for 100 labeled data with linear kernel.

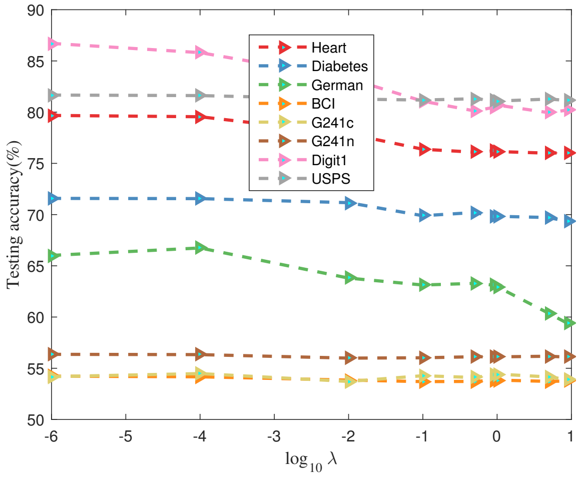

Figure 5. Accuracy for 10 labeled data with Gaussian kernel.

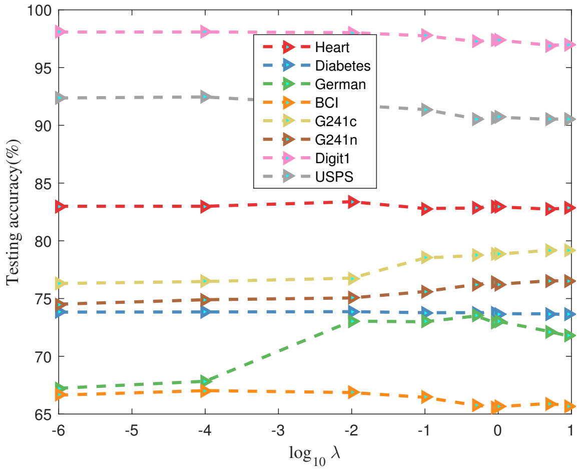

Figure 6. Accuracy for 100 labeled data with Gaussian kernel.

labeled and unlabeled samples. In comparison to RsLapRLS which uses ℓ_2 norm, the performance of the proposed algorithm is better and more robust. However, the proposed algorithm is designed to deal with two-class problems and multi-class problems often occur in some applications(i.e., face recognition). Additionally, compared to label information, pairwise constraints are more easily collected. Hence, how to design pairwise-constraint S3L methods and solve the multi-class problems will be the future work.

Acknowledgments

This work is supported by the High-level Talents Fund of Hubei University of Technology under grant No. GCRC2020016, Key Laboratory of Brain Machine Collaborative Intelligence of Zhejiang Province under grant No. 2020E10010-02, Open Research Fund Program of State Key Laboratory of Biocatalysis and Enzyme Engineering No. SKLBEE2020020, National Natural Science Foundation of China under grant No. 61601162, Shenzhen Institute of Artificial Intelligence and Robotics for Society.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. O. Chapelle, B. Scholkopf, A. Zien, editors, *Semi-Supervised Learning*, MIT Press, Cambridge, MA, 2006.
2. W. J. Chen, Y. H. Shao, C. N. Li, N. Y. Deng, MLTSVM: A novel twin support vector machine to multi-label learning, *Pattern Recognit.*, **52** (2016), 61–74.
3. I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, T. S. Huang, Semisupervised learning of classifiers: theory, algorithms, and their application to human-computer interaction, *IEEE Trans. Pattern Anal. Mach. Intell.*, **26** (2004), 1553–1566.

4. X. D. Wang, R. C. Chen, C. Q. Hong, Z. Q. Zeng, Z. L. Zhou, Semi-supervised multi-label feature selection via label correlation analysis with l1-norm graph embedding, *Image Vision Comput.*, **63** (2017), 10–23.

5. H. Gan, N. Sang, R. Huang, X. Tong, Z. Dan, Using clustering analysis to improve semi-supervised classification, *Neurocomputing*, **101** (2013), 290–298.

6. X. Zhu, *Semi-supervised learning literature survey*, Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.

7. Z. Qi, Y. Xu, L. Wang, Y. Song, Online multiple instance boosting for object detection, *Neurocomputing*, **74** (2011), 1769–1775.

8. B. Tan, J. Zhang, L. Wang, Semi-supervised elastic net for pedestrian counting, *Pattern Recognit.*, **44** (2011), 2297 – 2304.

9. Y. Cao, H. He, H. H. Huang, Lift: A new framework of learning from testing data for face recognition, *Neurocomputing*, **74** (2011), 916–929.

10. H. Gan, N. Sang, R. Huang, Self-training-based face recognition using semi-supervised linear discriminant analysis and affinity propagation, *J. Opt. Soc. Am. A*, **31** (2014), 1–6.

11. J. Richarz, S. Vajda, R. Grzeszick, G. A. Fink, Semi-supervised learning for character recognition in historical archive documents, *Pattern Recognit.*, **47** (2014), 1011–1020.

12. G. Tur, D. H. Tur, R. E. Schapire, Combining active and semi-supervised learning for spoken language understanding, *Speech Commun.*, **45** (2005), 171 – 186.

13. B. Varadarajan, D. Yu, L. Deng, A. Acero, Using collective information in semi-supervised learning for speech recognition, in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, IEEE, (2009), 4633–4636.

14. N. V. Chawla, G. Karakoulas, Learning from labeled and unlabeled data: An empirical study across techniques and domains, *J. Artif. Intell. Res.*, **23** (2005), 331–366.

15. H. Gan, Z. Luo, Y. Sun, X. Xi, N. Sang, R. Huang, Towards designing risk-based safe laplacian regularized least squares, *Expert Syst. Appl.*, **45** (2016), 1–7.

16. H. Gan, N. Sang, X. Chen, Semi-supervised kernel minimum squared error based on manifold structure, in *Proceedings of the 10th International Symposium on Neural Networks*, Berlin, Heidelberg, **7951** (2013), 265–272.

17. A. Singh, R. Nowak, X. Zhu, Unlabeled data: Now it helps, now it doesn't, *Adv. Neural Inf. Proc. Syst.*, **21** (2008), 1513–1520.

18. T. Yang, C. E. Priebe, The effect of model misspecification on semi-supervised classification, *IEEE Trans. Pattern Anal. Mach. Intell.*, **33** (2011), 2093–2103.

19. Y. F. Li, Z. H. Zhou, Improving semi-supervised support vector machines through unlabeled instances selection, in *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, AAAI Press, (2011), 500–505.

20. T. Joachims, Transductive inference for text classification using support vector machines, in *Proceedings of the Sixteenth International Conference on Machine Learning*, San Francisco, CA, **99** (1999), 200–209.

21. Y. F. Li, S. B. Wang, Z. H. Zhou, Graph quality judgement: A large margin expedition, in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, (2016), 1725–1731.

22. Y. Wang, S. Chen, Safety-aware semi-supervised classification, *IEEE Trans. Neural Networks Learn. Syst.*, **24** (2013), 1763–1772.

23. Y. Wang, Y. Meng, Z. Fu, H. Xue, Towards safe semi-supervised classification: Adjusted cluster assumption via clustering, *Neural Process. Lett.*, 2017.

24. Y. F. Li, Z. H. Zhou, Towards making unlabeled data never hurt, in *Proceedings of the 28th International Conference on Machine Learning*, Omnipress, (2011), 1081–1088.

25. T. F. Covoes, R. C. Barros, T. S. da Silva, E. R. Hruschka, A. C. P. L. F. de Carvalho, Hierarchical bottom-up safe semi-supervised support vector machines for multi-class transductive learning, *J. Inf. Data Manage.*, **4** (2013), 357–373.

26. H. Gan, Z. Li, W. Wu, Z. Luo, R. Huang, Safety-aware graph-based semi-supervised learning, *Expert Syst. Appl.*, **107** (2018), 243–254.

27. Y. F. Li, H. W. Zha, Z. H. Zhou, Learning safe prediction for semi-supervised regression, in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, San Francisco, California, (2017), 2217–2223.

28. H. Gan, Z. Li, Safe semi-supervised learning from risky labeled and unlabeled samples, in *2018 Chinese Automation Congress*, IEEE, (2018), 2096–2100.

29. M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: A geometric framework for learning from labeled and unlabeled examples, *J. Mach. Learn. Res.*, **7** (2006), 2399–2434.

AIMS Press