International Journal of Machine Learning and Cybernetics
https://doi.org/10.1007/s13042-018-0788-7

ORIGINAL ARTICLE

Adaptive safety degree-based safe semi-supervised learning

Nong Sang[1,2] · Haitao Gan[1] · Yingle Fan[1] · Wei Wu[1] · Zhi Yang[3]

Received: 23 April 2016 / Accepted: 20 January 2018

Abstract
Recently, safe semi-supervised learning has attracted more and more attention in the machine learning field. Many methods are introduced to safely exploit unlabeled data by designing different safe mechanisms. However, they assume that the risk or safety degrees are equal for all unlabeled data. In this paper, we propose an adaptive safe semi-supervised learning framework where the safety degrees of different unlabeled data are different and adaptively computed. In this framework, a safety degree-based tradeoff term between supervised learning (SL) and semi-supervised learning (SSL) is incorporated into the objective function of SSL. Then the optimal problem is solved by using an alternating iterative strategy. In particular, we utilize Regularized Least Squares (RLS) and Laplacian RLS (LapRLS) for SL and SSL, respectively. Our experimental results on several datasets demonstrate that the performance of our algorithm is never significantly inferior to that of RLS and LapRLS and show the effectiveness of our proposed safety mechanism.

Keywords Semi-supervised learning · Laplacian Regularized Least Squares · Adaptive safety degree · Safe mechanism

1 Introduction

In the past years, semi-supervised learning (SSL) [4, 26, 27] has been an interesting topic in the machine learning field and a number of methods [1, 8, 17, 24, 25] were proposed to make full use of both labeled and unlabeled data to obtain the better performance than supervised learning (SL). Generally speaking, SSL utilizes various assumptions to establish the relationship between labeled and unlabeled data, such as smoothness, cluster, and manifold assumption, etc. One of the most widely used assumptions is manifold assumption [2, 14]. Belkin [2] proposed Laplacian regularized Least Squares (LapRLS) and Support Vector Machines (LapSVM) by employing a graph regularization term to exploit the labeled and unlabeled data. The experimental results show the effectiveness of manifold assumption.

SSL, without doubt, can improve the performance with less labeled data in some cases and has been successfully applied to various fields, such as image classification [3, 7, 15, 18], object detection and tracking [5, 10, 16, 20], etc. Nevertheless, some researches [6, 19] have pointed out that the unlabeled data do not always help learn a semi-supervised classifier and may hurt the performance of SSL in some cases. This means that the unlabeled data may be risky and SSL even performs worse than SL only using the labeled data. It will be inimical to the practical applications for SSL to some extent. Therefore, It is expected to design a safe semi-supervised learning method that never performs worse than the corresponding supervised learning method. Li et al. [12] invented an S3VM_*us* method where a hierarchical clustering method was used to select the helpful unlabeled data. The selected helpful unlabeled data were then predicted by transductive SVM (TSVM) [11] and the remaining were predicted by SVM. Hence, the probability of performance degeneration is much smaller than that of TSVM. Simultaneously, Li et al. [13] developed a safe semi-supervised SVMs (S4VMs). Unlike S3VM which only tried to find one optimal classifier, S4VMs constructed multiple candidate low-density classifiers simultaneously to reduce the risk of selecting one poor classifier with unlabeled data. Finally, the performance of S4VMs is never significantly inferior to that of SVM. Afterwards, Wang et al. [23]

✉ Haitao Gan
ght1102@gmail.com

[1] School of Automation, Hangzhou Dianzi University, Hangzhou 310018, China

[2] School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China

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

Published online: 24 January 2018

developed a safety-aware SSCCM (SA-SSCCM) which is extended from semi-supervised classification method based on class membership (SSCCM). They designed a safe mechanism by controlling a tradeoff between least-square SVM (LS-SVM) and SSCCM. The results demonstrated that the performance of SA-SSCCM is not only never significantly inferior to that of LS-SVM, but seldom significantly inferior to that of SSCCM.

Though the above-mentioned methods considered the risk and could safely exploit the unlabeled data to some extent, they did not consider the different safety degrees of the unlabeled data. It is more reasonable to assume that different unlabeled data should have different safety degrees in learning a semi-supervised classifier. Based on this assumption, we propose an adaptive safe SSL framework. The basic idea is to adaptively assign a safety degree to each unlabeled data. Then a safety degree-based tradeoff term between SL and SSL is incorporated into the objective function of SSL. It is expected that the predictions of unlabeled data approach to that of SL when they may degenerate the performance of SSL. The reformulated optimal problem is finally resolved by using an alternating iterative strategy. In particular, we utilize Regularized Least Squares (RLS) and LapRLS for SL and SSL, respectively. The main contributions of the paper can be summarized as:

1. The different safety degrees of unlabeled data are introduced in our algorithm and a safety degree-based safe mechanism is designed to reduce the risk of the unlabeled data.
2. The safety degrees are adaptively computed in the iterative solution of our algorithm and it will avoid assigning the safety degrees artificially.

The rest of the paper is organized as follows: In Sect. 2, we firstly review the related work. Then we will give the details of our algorithm in Sect. 3. Section 4 will present a series of experiments on several datasets and analyze the results. Finally, we will conclude the paper in Sect. 5.

2 Related work

In this section, we will review the related work, including RLS and LapRLS.

2.1 Regularized Least Squares (RLS)

RLS is a kind of supervised learning which only uses labeled data to train a classifier [21, 22]. Given a labeled dataset $X = \{(x_1, y_1), \cdots, (x_l, y_l)\}$ with size l, $x_i \in \mathbb{R}^{\mathrm{D}}$ and $y_i \in \{-1, 1\}$. The objective function of RLS can be provided as:

$$\mathcal{J}(f) = \frac{1}{l}\sum_{i=1}^{l}(f(x_i) - y_i)^2 + \gamma\|f\|_K^2 \tag{1}$$

here, $\|\cdot\|_K$ is the norm defined in $\mathcal{H}_K$ which is a Reproducing Kernel Hilbert Space (RKHS) associated with a Mercer kernel $K : X \times X \to \mathbb{R}$.

According to the Representer Theorem, the solution can be given as

$$f(x) = \sum_{i=1}^{l}\alpha_i k(x_i, x) \tag{2}$$

By Substituting Eq.(2) into Eq.(1), the objective function can be rewritten as:

$$\mathcal{J}(\alpha) = \frac{1}{l}(K\alpha - Y)^T(K\alpha - Y) + \gamma\alpha^T K\alpha \tag{3}$$

where $\alpha = [\alpha_1, \cdots, \alpha_l]^T$, $Y = [y_1, \cdots, y_l]^T$, and K is the Gram matrix with size $l \times l$ whose entry $K_{ij} = K(x_i, x_j)$.

By setting the derivative of Eq.(3) to zero, we can obtain the optimal solution:

$$\alpha^* = (K + \gamma lI)^{-1}Y \tag{4}$$

where I is an identity matrix with size $l \times l$.

2.2 Laplacian Regularized Least Squares (LapRLS)

Given the labeled data X_l as above-mentioned, and the unlabeled data $X_u = \{x_{l+1}, \cdots, x_n\}$ with size $u = n - l$, $x_i \in \mathbb{R}^{\mathrm{D}}$. LapRLS employed the manifold assumption to learn from the labeled and unlabeled data. And LapRLS used a graph regularization term to exploit the manifold structure.

$$\mathcal{R} = \frac{1}{2}\sum_{i=1}^{n}(f(x_i) - f(x_j))^2 W_{ij} = f^T L f \tag{5}$$

where $f = [f(x_1), \cdots, f(x_n)]^T$, and L is the graph Laplacian defined as $L = D - W$. Here D is a diagonal matrix whose entry $D_{ii} = \sum_j W_{ij}$ and W is an edge weight matrix. The edge weight matrix can be computed as follows:

$$W_{ij} = \begin{cases} \exp\left\{-\frac{\|x_i - x_j\|_2^2}{2\sigma^2}\right\} & \text{if} \quad x_i \in N_p(x_j) \text{ or } x_j \in N_p(x_i) \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

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

By introducing the graph regularization term, the objective function of LapRLS can be written as follows:

$$\mathcal{J}_r(f) = \frac{1}{l}\sum_{i=1}^{l}(f(x_i)-y_i)^2 + \gamma_A\|f\|_K^2 + \frac{\gamma_I}{n^2}f^TLf \tag{7}$$

According to the Representer Theorem, we can rewrite the objective function as:

$$\begin{aligned}\mathcal{J}_r(\alpha) = &\frac{1}{l}\sum_{i=1}^{l}(GK\alpha-\tilde{Y})^T(GK\alpha-\tilde{Y}) + \gamma_A\alpha^TK\alpha \\ &+ \frac{\gamma_I}{n^2}\alpha^TKLK\alpha\end{aligned} \tag{8}$$

here, G is an $n\times n$ diagonal matrix where $G_{ii}=1$ for labeled data x_i and 0 for unlabeled data, K is the $n\times n$ Gram matrix over both labeled and unlabeled data whose entry $K_{ij}=K(x_i,x_j)$, $\tilde{Y}$ is a recoded label vector given by $\tilde{Y}=[y_1,\cdots,y_l,0,\cdots,0]^T$.

By setting the derivative of Eq. (8) to zero, we can obtain the optimal solution:

$$\alpha^* = \left(GK + \gamma_A lI + \frac{\gamma_I l}{n^2}LK\right)^{-1}Y \tag{9}$$

where I is an identity matrix with size $n\times n$.

3 Our algorithm

In this section, we will introduce our algorithm in details. Generally speaking, it is a reasonable assumption that the unlabeled data should have different safety degrees or impacts on learning a semi-supervised classifier. Our idea is to adaptively assign different safety degrees to unlabeled data and design a safe mechanism to exploit the unlabeled data. On the one hand, all the unlabeled data are used to learn a semi-supervised classifier and their predictions are restricted to approach to that of SL with different degrees. This will reduce the risk of unlabeled data. On the other hand, the safe unlabeled data should have larger safety degrees than the risky ones. Hence The safe mechanism is implemented through a safety degree-based tradeoff term between SL and SSL. In this way, our algorithm is a combination of SSL and SL. And the tradeoff term is then incorporated into the objective function of SSL. In particular, we utilize RLS and LapRLS as SL and SSL, respectively.

Based on the above analysis, we next propose an adaptive safety degree-based LapRLS method. For convenience, we denote the semi-supervised classifier obtained by our algorithm as $f(x)$ and supervised classifier obtained by RLS as $g(x)$. The reformulated objective function can be presented as follows:

$$\begin{aligned}\mathcal{Q}(f,s) = &\sum_{i=1}^{l}(f(x_i)-y_i)^2 + \gamma_A\|f\|_K^2 + \gamma_I f^TLf \\ &+ \lambda\sum_{j=l+1}^{n}s_j^m\|f(x_j)-g(x_j)\|_2^2 \\ s.t.\quad &\sum_{j=l+1}^{n}s_j = 1 \\ &0\le s_j\le 1, \forall j = l+1,\cdots,n.\end{aligned} \tag{10}$$

where s_j describes the safety degree of unlabeled data x_j and $m>1$ is a smoothness factor to control the degree of s_j. When $m\to\infty$, the last term will be ineffective and our algorithm will boil down to LapRLS. The parameter λ is a regularization parameter.

According to the Representer Theorem, we can rewrite the objective function as:

$$\begin{aligned}\mathcal{Q}(\alpha,s) = &(K_l\alpha-Y)^T(K_l\alpha-Y) + \gamma_A\alpha^TK\alpha + \gamma_I\alpha^TK \\ &\times LK\alpha + \lambda(K_u\alpha-K_{ul}\alpha')^TS(K_u\alpha-K_{ul}\alpha')\end{aligned} \tag{11}$$

where α' is the optimal solution of RLS and $K=\begin{bmatrix}K_l\\K_u\end{bmatrix}=\begin{bmatrix}K_{ll}&K_{lu}\\K_{ul}&K_{uu}\end{bmatrix}$. S is a diagonal matrix whose entry $S_{jj}=s_{j+l}^m$.

According to Ref. [9], the objective function $\mathcal{Q}(\alpha,s)$ is biconvex in (α,s). We can employ an alternating iterative strategy to solve the above optimal problem. First, when s_j is fixed, the derivative of Eq. (11) with respect to α is

$$\begin{aligned}\frac{\partial\mathcal{Q}}{\partial\alpha} = &K_l^T(K_l\alpha-Y) + \gamma_AK\alpha + \gamma_IKLK\alpha \\ &+ \lambda K_u^TS(K_u\alpha-K_{ul}\alpha')\end{aligned} \tag{12}$$

By setting Eq. (12) to zero, we can get

$$\begin{aligned}\alpha^* = &(K_l^TK_l + \gamma_AK + \gamma_IKLK + \lambda K_u^TSK_u)^{-1} \\ &\times(K_l^TY + \lambda K_u^TSK_{ul}\alpha')\end{aligned} \tag{13}$$

Then, when α is fixed, we can use the Lagrange multiplier method to find the solution. The relevant part can be written as:

$$\mathcal{L}(s_j) = \lambda\sum_{j=l+1}^{n}s_j^m\|f(x_j)-g(x_j)\|_2^2 - \eta\left(\sum_{j=l+1}^{n}s_j - 1\right) \tag{14}$$

By setting the derivative of Eq. (14) with respect to s_j to zero, the optimal value of s_j can be obtained.

$$s_j = \frac{\left(f(x_j) - g(x_j)\right)^{-\frac{2}{m-1}}}{\sum_{t=l+1}^{n} \left(f(x_t) - g(x_t)\right)^{-\frac{2}{m-1}}} \tag{15}$$

As can be seen from Eq. (15), if the difference $f(x_j) - g(x_j)$ is small, the corresponding unlabeled data x_j may be safe and the safety degree s_j should be large. Therefore, the safe unlabeled data have larger impacts on learning $f(x)$ than the risky ones. Additionally, a larger s_j will lead to a smaller difference $f(x_j) - g(x_j)$ by optimizing Eq.(10). Hence the predictions of unlabeled data will approach to that using SL and it will reduce the risk of the unlabeled data.

Finally, when we get the optimal value α^*, we can use $f(x) = \sum_{i=1}^{n} \alpha_i^* k(x_i, x)$ to classify the unseen data x.

The implementation of our algorithm can be described in Algorithm 1.

Algorithm 1 Our algorithm

Input: A labeled set $X_l = \{(x_1, y_1), \cdots, (x_l, y_l)\}$ and an unlabeled set $X_u = \{x_{l+1}, \cdots, x_n\}$, $n = l + u$, the parameters γ_A, γ_I, λ, p, M and ε.

Output: Decision coefficient α^*.

```
Initialize the safety degree s_j = 1/u;
Construct p nearest neighbor graph W and computer the Gram matrix K;
Compute the objective function value Q^(0) using Eq. (10);
Set t = 0;
while t ≤ M do
    Update the decision coefficient α using Eq. (13);
    Update the safety degree s_j using Eq.(15);
    Compute the objective function value Q^(t);
    if |Q^(t) − Q^(t−1)| ≤ ε then
        Stop the iteration process and return α*.
    else
        t = t + 1;
    end if
end while
```

Table 1 Description of the datasets

ID	Dataset	# Samples	# Features
1	BCI	400	114
2	G241c	1500	241
3	G241n	1500	241
4	Digit1	1500	241
5	USPS	1500	241

4 Experiments

In this section, a series of experiments on several benchmark datasets [4] are carried out to compare our algorithm with supervised learning (i.e., RLS, SVM and LS-SVM), semi-supervised learning (i.e., LapRLS, TSVM, SSCCM), and safe semi-supervised learning (i.e., S3VM_*us* [12], S4VMs [13], SA-SSCCM_*en* [23]). In each dataset, there are two configurations: (1) 10 data points are labeled; (2) 100 data points are labeled. And the rest constitute the unlabeled data. The details of the datasets are described in Table 1. We employ the classification accuracy on the unlabeled data to measure the performance of different methods.

In the experiments, the parameters C_1, C_2, λ_1, λ_2, η and ε in SVM, LS-SVM, TSVM, SSCCM, S3VM_*us*, S4VMs, SA-SSCCM_*en* are respectively fixed to 100, 0.1, 100, 0.1, 1 and 10^{-3} as the parameter settings in [23]. λ in SA-SSCCM_*en* is obtained by an ensemble strategy where λ is selected among $\{0, 0.2, 0.4, 0.6, 0.8, 1\}$. The regularization parameter γ_l in RLS is fixed to 0.05, and p in LapRLS and our algorithm is set to 5. γ_A, γ_I and λ in LapRLS, and our algorithm are obtained by leave-one-out cross validation in the case of 10 labeled data and by 5-fold cross validation in the case of 100 labeled data. The search region of γ_A and γ_I is $\{10^{-6}, 10^{-4}, 10^{-2}, 10^{-1}, 1, 10^1, 10^2\}$ and that of λ is $\{10^{-6}, 10^{-4}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^4, 10^6\}$. Both linear and Gaussian kernels are tested in the experiments, and the width parameter δ for computing the Gram matrix K is set to the average distance between the data points when 10 data points are labeled, and obtained by 5-fold across validation among $\{\frac{1}{4}\delta, \frac{1}{2}\delta, \delta, 2\delta, 4\delta\}$ in the case of 100 labeled data.

Tables 2 and 3 provide the classification accuracy of different methods. In each row, the bold value denotes the best result over each dataset. In each part which respectively denotes the results of linear and Gaussian kernel, the third-to-last row gives the average accuracy of each methods. Thesecond-to-last row provides the win/tie/loss (W/T/L) counts where semi-supervised learning performs better/comparable/worse than the corresponding supervised learning. The last row gives the W/T/L counts where safe semi-supervised learning performs better/comparable/worse than the corresponding semi-supervised learning.

As can be seen from these tables, we have following conclusions:

1. Firstly, LapRLS performs significantly worse than RLS on 3/3 and 0/3 out of 5 cases with linear/Gaussian kernels for 10 and 100 labeled data, respectively. It illustrates that unlabeled data may degenerate the performance of semi-supervised learning. Whereas our algorithm never performs significantly worse than RLS.

Table 2 Classification accuracy with ten labeled data points

Linear	RLS	SVM	LS-SVM	LapRLS	TSVM	SSCCM	S3VM_*us*	S4VMs	SA-SSCCM_*en*	Our algorithm
1	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	**57.5**±2.9
2	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	61.6±3.3
3	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	**60.1**±3.7
4	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	77.6±5.1
5	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	**82.0**±1.7
Average	65.9	63.7	63.7	62.5	67.0	63.6	63.8	63.6	65.0	**67.8**
Semi-supervised vs. cor. supervised: W/T/L				1/1/3	2/1/2	2/2/1	0/5/0	0/5/0	3/2/0	3/2/0
Safe semi-supervised vs. cor. semi-supervised: W/T/L							3/0/2	2/1/2	2/3/0	4/1/0
Gaussian	RLS	SVM	LS-SVM	LapRLS	TSVM	SSCCM	S3VM_*us*	S4VMs	SA-SSCCM_*en*	Our algorithm
1	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	**54.5**±2.1
2	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	58.1±3.2
3	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**±4.0
4	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	**86.4**±11.6
5	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	**83.4**±2.7
Average	63.7	59.0	58.3	61.7	63.0	60.0	59.5	62.8	63.4	**68.2**
Semi-supervised vs. cor. supervised: W/T/L				1/1/3	2/2/1	3/0/2	1/4/0	1/4/0	5/0/0	5/0/0
Safe semi-supervised vs. cor. semi-supervised: W/T/L							1/2/2	1/2/2	2/3/0	5/0/0

In each row, the bold values denote the best result over each dataset

Table 3 Classification accuracy with 100 labeled data points

Linear	RLS	SVM	LS-SVM	LapRLS	TSVM	SSCCM	S3VM_*us*	S4VMs	SA-SSCCM_*en*	Our algorithm
1	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	74.0±2.0
2	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	76.5±1.7
3	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	74.2±2.0
4	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	**92.6**±1.4
5	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	88.2±0.7
Average	76.9	80.0	80.6	74.3	81.4	80.2	80.0	80.5	**81.1**	**81.1**
Semi-supervised vs. cor. supervised: W/T/L				1/4/0	2/2/1	1/2/2	0/5/0	1/4/0	1/4/0	5/0/0
Safe semi-supervised vs. cor. semi-supervised: W/T/L							1/2/2	1/2/2	3/2/0	5/0/0
Gaussian	RLS	SVM	LS-SVM	LapRLS	TSVM	SSCCM	S3VM_*us*	S4VMs	SA-SSCCM_*en*	Our algorithm
1	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	69.4±2.4
2	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	77.6±2.8
3	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	**75.3**±2.0
4	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	95.2±1.8	**98.2**±0.5
5	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	87.8±2.1	**93.3**±1.8
Average	80.8	75.6	76.3	70.1	79.3	78.7	76.2	78.1	78.4	**82.8**
Semi-supervised vs. cor. supervised: W/T/L				1/1/3	3/1/1	3/1/1	1/4/0	2/3/0	3/2/0	3/2/0
Safe semi-supervised vs. cor. semi-supervised: W/T/L							1/1/3	1/2/2	1/3/1	4/1/0

In each row, the bold values denote the best result over each dataset

It indicates that our algorithm can be designed for safe semi-supervised learning.

2. Secondly, S3VM_*us* and S4VMs perform worse than TSVM totally in 9 and 8 cases out of 20 cases, respectively, and SA-SSCCM_*en* performs worse than SSCCM totally in 1 cases, whereas our algorithm never performs significantly worse than LapRLS. It illustrates that the safety degree-based safe mechanism is more effective and reasonable than that used in the other safe semi-supervised learning.
3. Finally, according to the average accuracy, our algorithm gives the comparable results with linear kernel for 100 labeled data and outperforms the other methods in the other three cases. Especially, the performance improve-

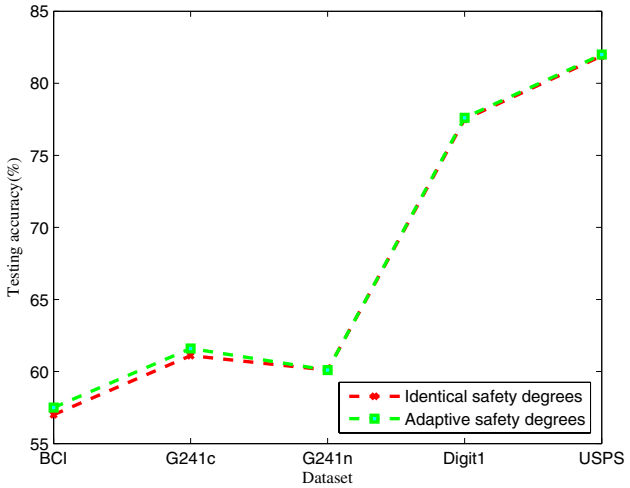

Fig. 1 The performance comparison between the identical and adaptive safety degrees of our algorithm for 10 labeled data with linear kernel

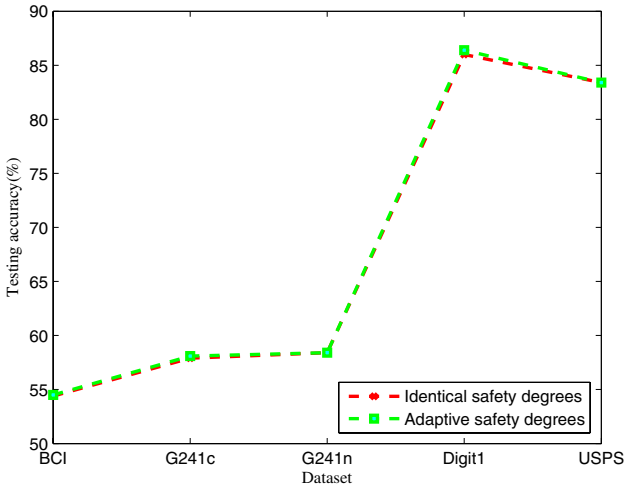

Fig. 3 The performance comparison between the identical and adaptive safety degrees of our algorithm for 10 labeled data with Gaussian kernel

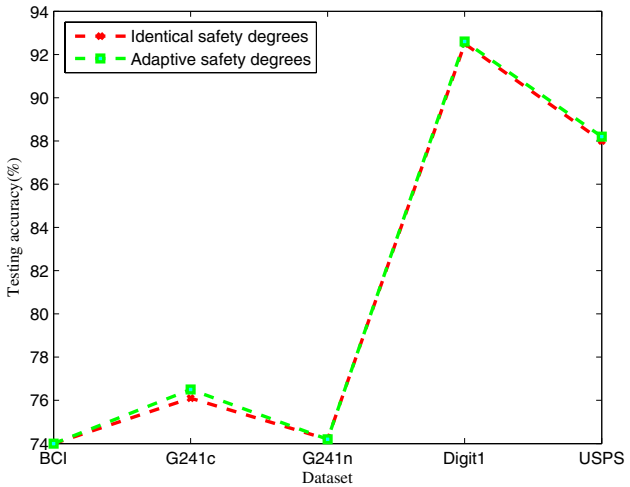

Fig. 2 The performance comparison between the identical and adaptive safety degrees of our algorithm for 100 labeled data with linear kernel

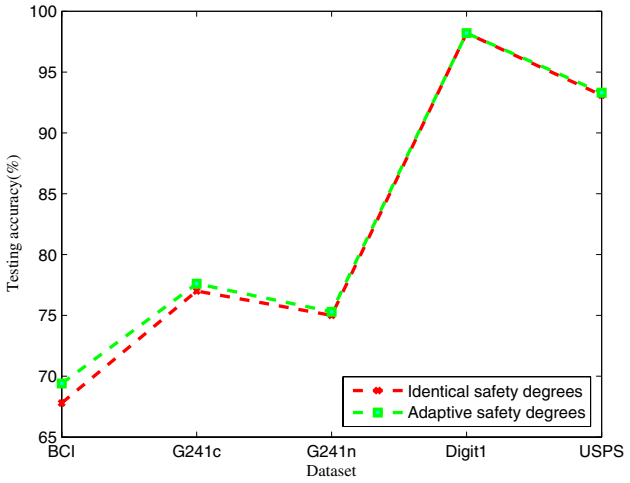

Fig. 4 The performance comparison between the identical and adaptive safety degrees of our algorithm for 100 labeled data with Gaussian kernel

ment of our algorithm is distinct in the case of 10 labeled data where there are little labeled data. It further shows the efficiency of the adaptive safe mechanism used in our algorithm.

Furthermore, we investigate the performance impact of the adaptive safety degrees used in our algorithm. Figs. 1, 2, 3, 4 show the results of two kinds of safety degrees. Green lines denotes the performance of our algorithm using the adaptive safety degree. while red lines denotes the performance of our algorithm using the identical safety degree of different unlabeled data where $s_j = \frac{1}{u}$. From these figures, we can find that our algorithm using the adaptive safety degree can perform better, even slightly, than that using the identical safety degree in all cases. It further explains why we consider the different safety degrees of the unlabeled data. Additionally, we analyze the impact of λ on the performance of our algorithm. The value of λ is selected among $\{10^{-6}, 10^{-4}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^4, 10^6\}$. Fig. 5, 6,

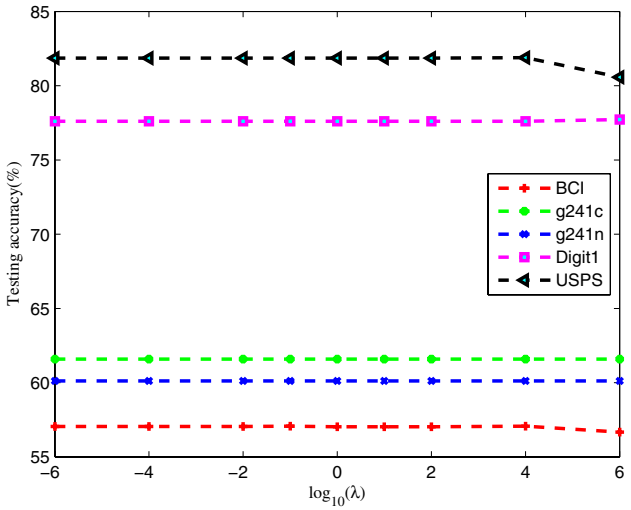

Fig. 5 Accuracy for 10 labeled data with linear kernel

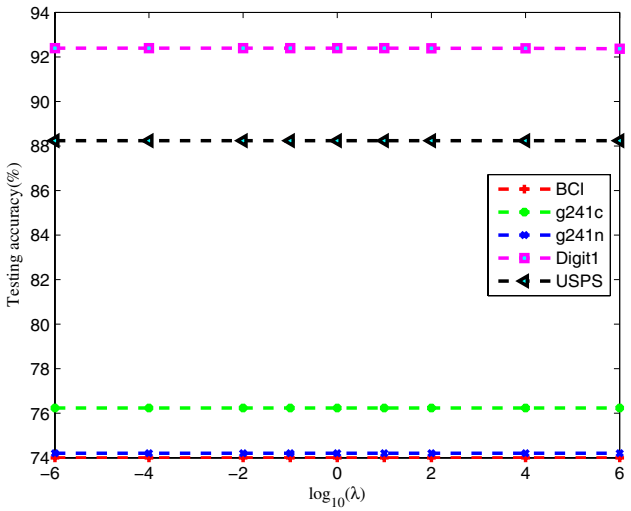

Fig. 6 Accuracy for 100 labeled data with linear kernel

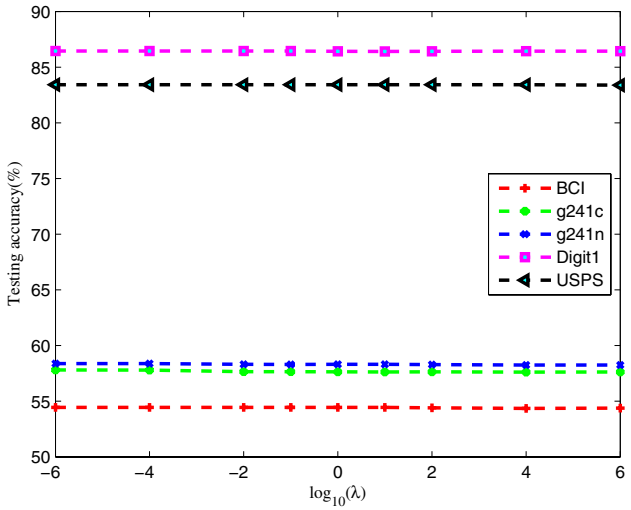

Fig. 7 Accuracy for 10 labeled data with Gaussian kernel

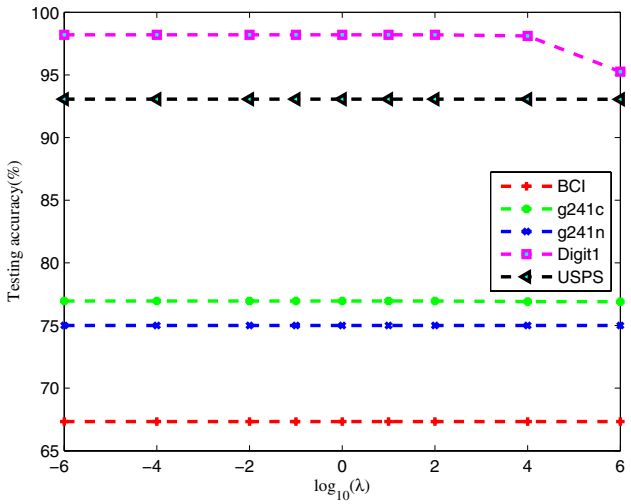

Fig. 8 Accuracy for 100 labeled data with Gaussian kernel

7, 8 give the performance of our algorithm as the parameter λ changes. As can be seen, the performance is relatively stable when the value of λ is small and will decease on BCI, Digit1 and USPS datasets when the value is greater than 10^4. It is indicated that one can easily find the optimal interval of λ. This shows the advantage of our algorithm over SA-SSCCM_*en* which is sensitive to the parameter λ.

5 Conclusion

In this paper, we propose a novel mechanism to design a safe semi-supervised learning method based on RLS and LapRLS. Our motivation is that unlabeled data should have different safety degrees to learn a semi-supervised classifier. In order to consider the safety degree, we employ an adaptive generated way to obtain the safety degrees of different unlabeled data. And then a safety degree-based tradeoff term between RLS and LapRLS is incorporated into the objective function of LapRLS. The experimental results on several benchmark datasets demonstrate that the performance of our algorithm is never significantly inferior to that of RLS and LapRLS and indicate the effectiveness of our designed safe mechanism.

Acknowledgements Funding: This work is funded by Natural Science Foundation of China under Grants Nos. 61601162, 61671197, 61501154 and 60872090, and Open Foundation of first level Zhejiang key in key discipline of Control Science and Engineering, and Zhejiang Province Education Department under Grant Y201328513.

References

1. Adankon MM, Cheriet M (2010) Genetic algorithm-based training for semi-supervised svm. Neural Comput Appl 19(8):1197–1206
2. Belkin M, Niyogi P, Sindhwani V (2006) Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J Mach Learn Res 7:2399–2434
3. Cao Y, He H, Huang H (2011) Lift: a new framework of learning from testing data for face recognition. Neurocomputing 74(6):916–929
4. Chapelle O, Scholkopf B, Zien A (eds) (2006) Semi-supervised learning. MIT Press, Cambridge
5. Chen S, Li S, Su S, Cao D, Ji R (2014) Online semi-supervised compressive coding for robust visual tracking. J Vis Commun Image Represent 25(5):793–804
6. Cozman FG, Cohen I, Cirelo MC, Politecnica E (2003) Semi-supervised learning of mixture models. In: Proceedings of the 20th International Conference on Machine Learning. Omnipress, Madison, pp 99–106
7. Gan H, Sang N, Huang R (2014) Self-training-based face recognition using semi-supervised linear discriminant analysis and affinity propagation. J Opt Soc Am A Optics, Image Sci Vis 31(1):1–6
8. Gan H, Sang N, Huang R, Tong X, Dan Z (2013) Using clustering analysis to improve semi-supervised classification. Neurocomputing 101:290–298
9. Gorski J, Pfeuffer F, Klamroth K (2007) Biconvex sets and optimization with biconvex functions: a survey and extensions. Math Methods Oper Res 66(3):373–407
10. Grabner Helmut LC, Horst B (2008) Semi-supervised on-line boosting for robust tracking. In: Proceedings of the 10th European Conference on Computer Vision: Part I. Springer-Verlag, Berlin, Heidelberg, pp 234–247
11. Joachims T (1999) Transductive inference for text classification using support vector machines. In: Proceedings of the Sixteenth International Conference on Machine Learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp 200–209
12. Li YF, Zhou ZH (2011) Improving semi-supervised support vector machines through unlabeled instances selection. In: Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence. AAAI Press, San Francisco, pp 500–505
13. Li YF, Zhou ZH (2011) Towards making unlabeled data never hurt. In: Proceedings of the 28th International Conference on Machine Learning, Omnipress, Madison, pp 1081–1088
14. Liu B, Xia SX, Meng FR, Zhou Y (2016) Manifold regularized extreme learning machine. Neural Comput Appl 27(2):255–269
15. Lu Z, Wang L (2015) Noise-robust semi-supervised learning via fast sparse coding. Pattern Recogn 48(2):605–612
16. Qi Z, Xu Y, Wang L, Song Y (2011) Online multiple instance boosting for object detection. Neurocomputing 74(10):1769–1775
17. Reddy IS, Shevade S, Murty M (2011) A fast quasi-newton method for semi-supervised SVM. Pattern Recogn 44(10–11):2305–2313
18. Richarz J, Vajda S, Grzeszick R, Fink GA (2014) Semi-supervised learning for character recognition in historical archive documents. Pattern Recogn 47(3):1011–1020
19. Singh A, Nowak R, Zhu X (2009) Unlabeled data: Now it helps, now it doesn't. In: Koller D, Schuurmans D, Bengio Y, Bottou L (eds) Advances in neural information processing systems, vol 21. Curran Associates Inc, Red Hook, pp 1513–1520
20. Tan B, Zhang J, Wang L (2011) Semi-supervised elastic net for pedestrian counting. Pattern Recogn 44(10–11):2297–2304
21. Wang R, Wang XZ, Kwong S, Xu C (2017) Incorporating diversity and informativeness in multiple-instance active learning. IEEE Trans Fuzzy Syst 25(6):1460–1475
22. Wang XZ, Wang R, Feng HM, Wang HC (2014) A new approach to classifier fusion based on upper integral. IEEE Trans Cybern 44(5):620–635
23. Wang Y, Chen S (2013) Safety-aware semi-supervised classification. IEEE Trans Neural Netw Learn Syst 24(11):1763–1772
24. Zhang Z, Zhen L, Deng N, Tan J (2015) Manifold proximal support vector machine with mixed-norm for semi-supervised classification. Neural Comput Appl 26(2):399–407
25. Zhou ZH, Li M (2005) Tri-training: exploiting unlabeled data using three classifiers. IEEE Trans Knowl Data Eng 17(11):1529–1541
26. Zhu H, Wang X (2017) A cost-sensitive semi-supervised learning model based on uncertainty. Neurocomputing 251(Supplement C):106–114
27. Zhu X, Goldberg AB (2009) Introduction to semi-supervised learning. Synth Lectures Artif Intell Mach Learn 3(1):1–130