Neural Process Lett (2017) 46:1031–1042
DOI 10.1007/s11063-017-9607-5

Towards Safe Semi-supervised Classification: Adjusted Cluster Assumption via Clustering

Yunyun Wang[1,2] · **Yan Meng**[1,2] · **Zhenyong Fu**[1,2] · **Hui Xue**[3]

Published online: 8 April 2017

Abstract Semi-supervised classification methods can perform even worse than the supervised counterparts in some cases. It undoubtedly reduces their confidence in real applications, and it is desired to improve the safety of semi-supervised classification such that it never performs worse than the supervised counterpart. Considering that the cluster assumption may not well reflect the real data distribution, which can be one possible cause of unsafe learning, we develop a safe semi-supervised support vector machine method in this paper by adjusting the cluster assumption (ACA-S3VM for short). Specifically, when samples from different classes are seriously overlapped, the real boundary actually lies not in the low density region, which will not be found by the cluster assumption. However, an unsupervised clustering method is able to detect the real boundary in this case. As a result, we design ACA-S3VM by adjusting the cluster assumption with the help of clustering, which considers the distances of individual unlabeled instances to the distribution boundary in learning. Empirical results show the competition of ACA-S3VM compared with the off-the-shelf safe semi-supervised classification methods.

Keywords Semi-supervised classification · Cluster assumption · Clustering · Decision boundary · Low density region

1 Introduction

In many real applications, such as web page recommendation and spam email detection, the unlabeled data can be easily and cheaply collected, while the acquisition of labeled

✉ Yunyun Wang
wangyunyun@njupt.edu.cn

1 Department of Computer Science and Engineering, Nanjing University of Posts and Telecommunications, 210046 Nanjing, People's Republic of China

2 Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing 210023, Jiangsu, People's Republic of China

3 School of Computer Science and Engineering, Southeast University, Nanjing 210096, People's Republic of China

data is usually quite expensive and time-consuming, especially involving manual effort. Consequently, semi-supervised learning, which exploits both labeled and unlabeled data for learning, has attracted intensive attention during the past decades. In this paper, we focus on semi-supervised classification, and lots of semi-supervised classification methods have been developed so far [1–4].

Generally, semi-supervised classification methods attempt to exploit the intrinsic data distribution information disclosed by the unlabeled data in learning [5,6]. To exploit the unlabeled data, two distribution assumptions are usually adopted, i.e., the cluster assumption and the manifold assumption [3,4,7] The former assumes that similar instances are likely to share the same class label, thus guides the classification boundary passing through the low density region between clusters, thus it is also known as low density separation assumption. The latter assumes that data are resided on some low dimensional manifold represented by a Laplacian graph, and similar instances should share similar classification outputs according to the graph. Almost all off-the-shelf semi-supervised classification methods adopt one or both of those assumptions explicitly or implicitly [1,4]. For instance, the large margin semi-supervised classification methods, such as semi-supervised SVM (S3VM) [8] and the variants [9,10], adopt the cluster assumption. The graph-based semi-supervised classification methods, such as label propagation [11,12] and manifold regularization (MR) [13], adopt the manifold assumption.

Despite of the great enthusiasm in developing novel semi-supervised classification methods, however, it has been found that they may yield even worse performances than their supervised counterparts in some cases, or in other words, the unlabeled data may hurt the performance [14,15]. It undoubtedly reduces the confidence of adopting semi-supervised classification methods in real applications, and consequently, it is desired to develop safe semi-supervised classification methods never performing worse than the supervised counterparts. However, to the best of our knowledge, there are few researches [14–16] on safe semi-supervised classification up to now. Considering that not all unlabeled instances are helpful for learning, Li et al. developed the S3VM-*us* in [14] through selecting the unlabeled instances by hierarchical clustering. Specifically, only the unlabeled instances with high confidence by hierarchical clustering are predicted by TSVM, while the rest are predicted by SVM. Finally, the chance of its performance degeneration is much smaller than that of S3VM. At the same time, in [15,17], Li et al. developed the safe S3VM (S4VM) method. Different from S3VM seeking an optimal low-density separator, S4VM exploits the candidate low-density separators simultaneously to reduce the risk of identifying a poor separator with the unlabeled data. The performance of S4VM is highly competitive to S3VM and never significantly inferior to SVM. Both S3VM_*us* and S4VM work in the transductive learning style. In [16], Wang et al. invented a safety-control mechanism for safe semi-supervised classification by adaptive trade-off between semi-supervised and supervised classification in terms of unlabeled data, and further developed a safety-aware SSCCM (SA-SSCCM, safety-aware semi-supervised classification method based on class memberships [18]). The performance of SA-SSCCM is never worse than its supervised counterparts LS-SVM, and rarely worse than its corresponding semi-supervised counterparts SSCCM as well.

To address the issue of unsafe semi-supervised classification learning, we should first consider the cause of unsafe learning. Actually, there are mainly two possible causes: (1) The unlabeled instances can be unreliable so that they may mislead the classification. (2) The distribution assumption adopted may not well reflect the real data distribution so that it can mislead the classification. We follow the second line to consider the distribution assumption, and further, we focus on the cluster assumption, or low density assumption here. It states that "the classification should pass through the low density region". However, as shown in Fig. 1,

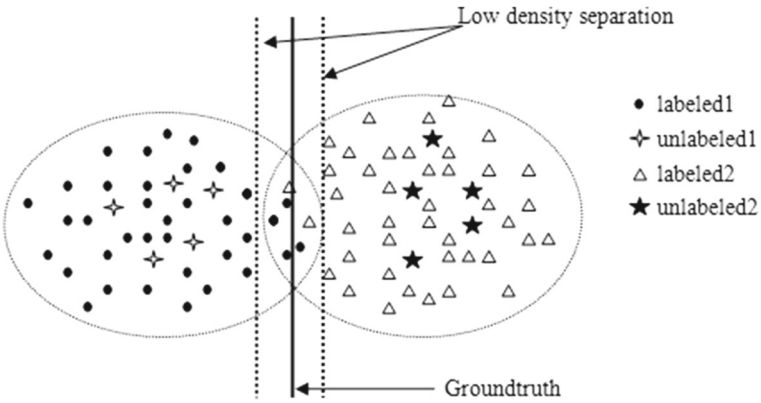

Fig. 1 An illustration in which the low density boundary is not the groundtruth

when samples from different classes are seriously overlapped, the real boundary actually lies not in the low density region. In this case, adopting the cluster assumption may lead to unsafe learning. However, a traditional unsupervised clustering method is able to generate a decision boundary close to the ground truth in this case. As a result, we attempt to adjust the cluster assumption to better fit the real data distribution with the help of clustering, and develop a new S3VM method based on the adjusted cluster assumption. Specifically, we first compute the distances of each unlabeled instances to the boundary in terms of clustering, and then incorporate them into S3VM so that instances in the cluster boundary will also be classified into the class boundary, in this way, the class boundary is guided to pass through the cluster boundary rather than the low-density region in Fig. 1. At the same time, when classes or clusters are not seriously overlapped, the real boundary lies in the low density region. In this case, the clustering boundary is also in the low density region, and thus the class boundary detected will be in the low density separation as well with no adjustment. In this way, for the cluster-assumption based methods, the decision boundary can be adjusted by clustering when different classes have serious overlaps such that the decision boundary does not lie in the low-density region [19]. It can undoubtedly alleviate the second cause of unsafe semi-supervised classification to some extent. Further, the distances of instances to the boundary actually describe their confidence in classification, thus the new method considers the confidence of individual unlabeled instances in classification. As a result, it may alleviate the first cause as well. As a result, it is respected to improve the safety of semi-supervised classification.

It is easily noted that S3VM_*us* adopts a similar strategy of considering the individual unlabeled instances for safe semi-supervised learning. However, S3VM_*us* stems from instance selection, and learns in the transductive learning style, while ACA-S3VM starts from adjusting the cluster assumption, and learns in the inductive learning style. Moreover, each unlabeled instance in S3VM_*us* is either selected (predicted by S3VM) or discarded (predicted by SVM) by some confidence threshold, thus in both the selected and the discarded datasets, different confidences of individual unlabeled instances are actually discarded. While through considering the different distances of unlabeled instances to the boundary, different confidences of individual unlabeled instances are actually utilized for further improving the safety of semi-supervised classification. Research in [19] also considers to improve the clus-

ter assumption is this case, however, the new method adopts a distribution assumption more like the manifold assumption considering the pairwise similarity between instances.

In the implementation, we take LS_S3VM for the base classifier, and develop a new safe LS_S3VM method based on Adjusted Cluster Assumption (ACA-S3VM for short). However, the proposed idea can be easily applied to other semi-supervised methods such as S3VM and LapRLSC, etc., and it can also be combined with other safe semi-supervised classification methods, such as S4VM or SA-SSCCM.

The rest of the paper is organized as follows: Sect. 2 introduces the related work, Sect. 3 describes the proposed ACA-S3VM method, Sect. 4 presents the empirical results and some conclusions are drawn in Sect. 5.

2 Related Works

In this paper, we aim to develop a safe semi-supervised classification method by adjusting the cluster assumption, and in the implementation, we adopt the semi-supervised LS-S3VM as the base classifier. As a result, we will briefly introduce the semi-supervised LS-S3VM, and the safe semi-supervised S3VM_*us* for comparison as well.

Given labeled data $X_l = \{x_i\}_{i=1}^{n_l}$ with corresponding labels $Y = \{y_i\}_{i=1}^{n_l}$, and unlabeled data $X_u = \{x_j\}_{j=n_l+1}^{n}$ where each $x_i \in R^d$ and $n_u = n - n_l$. With a decision function $f(x)$, the large margin semi-supervised classification method can be established in terms of the following formulation:

$$\min_{f,\hat{y}_j} \frac{1}{2}\|f\|^2 + \frac{C_1}{2l}\sum_{i=1}^{l} V\left(x_i, y_i, f(x_i)\right) + \frac{C_2}{2(n-l)}\sum_{j=l+1}^{n} V\left(x_j, \hat{y}_j, f(x_j)\right) \quad (1)$$

where $V(\cdot,\cdot,\cdot)$ is the loss function for classification, each $\hat{y}_j$ denotes the predicted label for the unlabeled instance x_j, $j = l+1 \ldots n$, C_1 and C_2 are the regularization parameters balancing the correct classifications between the labeled and unlabeled data, respectively. It can easily be found that the large margin semi-supervised method seeks for the discriminative function and the class labels for the given unlabeled instances simultaneously. When adopting the hinge loss [20] and square loss, respectively, we can get the famous S3VM [8,21] and LS-S3VM from (1).

Specifically, through adopting the square loss function for simply solution, the optimization problem of LS_S3VM can be formulated as:

$$\min_{f,\hat{y}_j} \frac{1}{2}\|f\|^2 + \frac{C_1}{2l}\sum_{i=1}^{l} \left(f(x_i) - y_i\right)^2 + \frac{C_2}{2(n-l)}\sum_{j=l+1}^{n} \left(f(x_j) - \hat{y}_j\right)^2 \quad (2)$$

LS_S3VM seeks for the decision function and the class labels for the unlabeled instances simultaneously. For the problem solving, the alternative iteration strategy can be used for obtaining the decision function and the predicted labels, respectively, in which each step generates a closed-form solution.

Based on S3VM, Li et al. developed a safe semi-supervised classification method S3VM_*us* through unlabeled instances selection. Specifically, in S3VM_*us*, both supervised SVM and semi-supervised S3VM are first performed independently to find the prediction inconsistent subset of the unlabeled data. Then a hierarchical clustering method is performed to calculate the instance confidence for this subset. Finally, only the unlabeled instances with high confidence in this subset are predicted by S3VM, while the rests are predicted by SVM. As a result, S3VM_*us* seeks only the class labels for the given unlabeled instances, thus

learns in the transductive style. Further, the unlabeled instances is either selected (predicted by S3VM) or discarded (predicted by SVM) by some confidence threshold, thus in both the selected and the discarded datasets, different confidences of individual instances are actually discarded. In this paper, we aim to develop a safe semi-supervised classification method, which is inductive, and moreover, different confidences of individual unlabeled instances can be considered.

3 Safe Semi-supervised Classification by Adjusted Cluster Assumption

When different classes have seriously overlaps, the distribution boundary does not lie around the low-density region. As a result, semi-supervised methods adopting the cluster assumption, such as S3VM, may lead to unsafe learning in this case. Considering this issue, we attempt to develop a new semi-supervised classification method in this sub-section by adjusting the cluster assumption. Further, a traditional clustering method is able to detect the real class boundary in this case, thus we will adopt clustering to adjust the cluster assumption, and develop a corresponding semi-supervised classification method. We will give the mode description, problem solving and algorithm description in separated sub-sections, respectively.

3.1 Mode Description

Before describing the formulation of ACA-S3VM, we first introduce a distance vector $\mathbf{V} \in R^n$, which is used to describe the distances of individual instances to the class boundary. Each entry V_i in $\mathbf{V}$ is defined as $V_i = |d(x_i - v_1) - \mathrm{d}(x_i - v_2)|$, where $|\cdot|$ is used to get the absolute value, v_1 and v_2 are the cluster centres by a pre-performed fuzzy clustering method such as FCM, and $d(x_i - v_k)$ denotes the distance of each x_i to the kth cluster ($k = 1$ or 2). Further, $\mathbf{V}$ is normalized to be a normal distribution with centre 1, and moreover, each V_i is set to be 0 if $V_i < 0$. As a result, when V_i is large, instance x_i is far from the cluster boundary, otherwise, x_i is more likely to lie in the cluster boundary.

Assuming that instances in the cluster boundary also lie in the class boundary, the optimization problem of ACA-S3VM can be formulated as

$$\min_{f,\hat{y}_j} \frac{1}{2}\|f\|_K^2 + \frac{C_1}{2l}\sum\nolimits_{i=1}^{l}(f(x_i) - y_i)^2 + \frac{C_2}{2(n-l)}\sum\nolimits_{j=l+1}^{n}\left(f(x_j) - \hat{y}_j V_j\right)^2 \quad (3)$$

The distance vector $\mathbf{V}$ is incorporated into the third item of (3). When V_j approaches 0, the unlabeled instance x_j is likely to lie in the cluster boundary, then its prediction $f(x_j)$ is restricted to be close to 0 in (3). It actually guides x_j to be classified into the area of class boundary. Otherwise, when V_j is large, x_j is far from the cluster boundary, then it is restricted to be far from the decision boundary by (3).

For the cluster-assumption based methods, the clusters may have serious overlaps such that the decision boundary does not lie in the low-density region [19]. Through incorporating the cluster structure in ACA-S3VM, the decision boundary can be adjusted to the real boundary. At the same time, when classes or clusters are not seriously overlapped, the real boundary does lie in the low density region. In this case, the class boundary detected will be in the low density separation with no adjustment, since the clustering boundary is also in the low density region. As a result, it can undoubtedly alleviate the impact of mis-specified distribution assumption to some extent. At the same time, the distances of instances to the boundary actually describe their confidence in classification, as a result, ACA-S3VM considers the confidence of individual unlabeled instances in classification, thus it may also alleviate the

impact of misleading unlabeled instances to some extent. As a result, it is respected to improve the safety of semi-supervised classification.

Note that we take LS-S3VM as the base classifier, or adopt the square loss function here due to its simplicity in problem solving, however, the strategy can also be applied to other semi-supervised classifiers adopting other loss functions.

3.2 Problem Solving

Similar to LS-S3VM, ACA-S3VM seeks the discriminative function and the class labels for given unlabeled instances simultaneously. The optimization problem can be solved by an alterative iterating strategy to obtain $f(x)$ and $\hat{y}_j$s respectively. Specifically, with fixed $\hat{y}_j$s, the optimization problem for $f(x)$ can be formulated as

$$\min_f \frac{1}{2}\|f\|^2 + \frac{C_1}{2l}\sum_{i=1}^{l}(f(x_i)-y_i)^2 + \frac{C_2}{2(n-l)}\sum_{j=l+1}^{n}\left(f(x_j)-\hat{y}_j V_j\right)^2 \tag{4}$$

The minimizer of (4) has the form $f(x)=\sum_{i=1}^{n}\alpha_i^K(x_i,x)$ based on the Representer Theorem, then (4) can be further reformulated as

$$\begin{aligned}\min_\alpha J = {} & \frac{1}{2}\alpha^T K\alpha + \frac{C_1}{2l}(K_l\alpha - Y_l)^T(K_l\alpha - Y_l) \\ & + \frac{C_2}{2(n-l)}(K_u\alpha - Y_u V)^T(K_u\alpha - Y_u V)\end{aligned} \tag{5}$$

where $\alpha=[\alpha_1,\alpha_2\ldots\alpha_n]$ is the Lagrange multiplier vector. $K_{ll}=\langle\phi(X_l),\phi(X_l)\rangle_{\mathcal{H}}$, $K_{lu}=\langle\phi(X_l),\phi(X_u)\rangle_{\mathcal{H}}$and $K_{uu}=\langle\phi(X_u),\phi(X_u)\rangle_{\mathcal{H}}$ are kernel matrices, and $K=[K_lK_u]=\begin{bmatrix}K_{ll}K_{lu}\\K_{ul}K_{uu}\end{bmatrix}$. $\hat{V}\in R^{u\times u}$ is a diagonal matrix with each diagonal element $\hat{V}_{jj}$ being V_j.

By zeroing the derivative of J with respect to α, we have

$$\alpha=\left(K+\frac{C_1}{l}K_l^TK_l+\frac{C_2}{n-l}K_u^TK_u\right)^{-1}\left(\frac{C_1}{l}K_l^TY_l+\frac{C_2}{n-l}K_u^TY_uV\right) \tag{6}$$

At the same time, with fixed $f(x)$, the optimization problem for $\hat{y}_j$s can be written as

$$\min_{\hat{y}_j}\sum_{j=l+1}^{n}V_j\left(f(x_j)-\hat{y}_jV_j\right)^2 \tag{7}$$

Since $V_j >= 0$, thus for each x_j, if $f(x_j)\geq 0$, $\hat{y}_j=1$, else $\hat{y}_j=-1$.

3.3 Algorithm Description

The initial values for the class labels of the given unlabeled data are obtained by supervised LS-SVM, and the iteration terminates when $|M^k-M^{k-1}|<\varepsilon M^{k-1}$, where M^k denotes the objective function value at the kth iteration, and ε is a pre-defined threshold. The algorithm description of ACA-S3VM is summarized in Table 1.

Proposition 1 *The sequence* $\{J(\alpha_k,y_k)\}$ *obtained in the above algorithm w.r.t. ACA-S3VM converges.*

Table 1 The algorithm description of ACA-S3VM

Input	X_l, X_u --- the labeled and unlabeled data
	Y_l --- the labels of X_l
	C_1, C_2 -- the regularization parameters
	ε --- the iterative stop parameter
	Maxiter --- the maximum number for iteration
Output	$f(x)$, $\hat{y}_j$s --- the decision function and the labels for X_u

Procedure

Get the cluster centres by FCM, and then compute vector V;

Get the initial $\hat{y}_j$s of X_u by LS-SVM;

Set the initial objective function value to infinity, i.e., M_0=INF;

For k=1…Maxiter

  Update α by (6), and $f(x)$ by the Represent theorem with obtained α;

  Update $\hat{y}_j$s, and the objective function value M^k;

  If $|M^k\text{-}M^{k-1}| < \varepsilon M^{k-1}$

    Break, return $f(x)$ and $\hat{y}_j$s;

  Endif

Endfor

Proof First, the sequence of the objective function values generated by the above algorithm decreases monotonically. In fact, the objective function $J(\alpha, y)$ is biconvex [22] in (α, y). Specifically, for fixed y_k, the objective function is convex in α, thus the optimal α^* can be obtained by minimizing $J(\alpha, y_k)$, or equivalently optimizing (4). Now set $\alpha_{k+1} = \alpha^*$, then $J(\alpha_{k+1}, y_k) = J(\alpha^*, y_k) \leq J(\alpha_k, y_k)$. Simultaneously, with current α_{k+1}, the objective function is convex in y, thus the optimal y^* can be obtained by minimizing $J(\alpha_{k+1}, y)$, or equivalently optimizing (7). Now set $y_{k+1} = y^*$, then $J(\alpha_{k+1}, y_{k+1}) = J(\alpha_{k+1}, y^*) \leq J(\alpha_{k+1}, y_k)$. Finally, $J(\alpha_{k+1}, y_{k+1}) \leq J(\alpha_{k+1}, y_k) \leq J(\alpha_k, y_k), \forall k \in N$. Hence, the consequence $\{J(\alpha_k, y_k)\}$ decreases monotonically.

Further, since the objective function is non-negative, thus lower-bounded, as a result, the sequence $\{J(\alpha_k, y_k)\}$ converges. □

4 Experiments

To evaluate our ACA-S3VM, we perform comparison with supervised LS-SVM, semi-supervised LS-S3VM, LapRlsc, and the safe semi-supervised classification method S3VM_*us* over 7 UCI datasets[1] and 5 benchmark datasets.[2] The description of those datasets is given in Table 2. For comparison with transductive S3VM_*us*, we implement all methods in the transductive learning style, since inductive methods can predict the given unlabeled data as well by the decision function. Specifically, we report the prediction accuracy on the unlabeled instances available, although our method can directly achieve out-of-sample extension to predict unseen testing instances not adopted in learning.

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

[2] http://www.kyb.tuebingen.mpg.de/ssl-book/.

Table 2 Description of the 9 UCI datasets, including the number of instances and features

Dataset	#Dimension	#Instance		
		#positive	#negative	#total
automobile	25	71	88	159
house	16	168	267	435
ionosphere	34	225	126	351
sonar	60	97	111	208
wdbc	30	212	357	569
german	30	700	300	1000
isolet	51	300	300	600
BCI	241	750	750	1500
digit1	241	734	766	1500
g241c	241	750	750	1500
g241n	241	748	752	1500
USPS	241	300	1200	1500

4.1 Comparision Results

For the UCI datasets, each one is randomly split into two halves, one for training and the other for testing, and the training set contains 10 and 100 labeled instances, respectively, with the rests unlabeled. This process along with the classifier learning is repeated 20 times, and the average accuracy and variance are reported. For each benchmark dataset, there are two settings, one including 10 labeled instances and the other including 100 instances. Further, for each setting, there are 12 subsets of labeled data and unlabeled instances partitions, finally the average prediction performances on the unlabeled data are reported.

The Gaussian kernel is adopted here, and the width parameter in the Gaussian kernel is fixed to the average distance between all instance pairs. Each regularization parameter is selected from $\{0.01, 0.1, 1, 10, 100\}$. When 10 instances are labeled, the best performance over all parameter combinations is reported. When 100 instances are labeled, the results are reported with parameters selected by cross-validation. The results are reported in Tables 3 and 4, where the bold value in each row indicates the best performance over each dataset, the superscript "*" indicates that ACA-S3VM performs better than supervised SVM, and subscript "*" indicates that ACA-S3VM performs better than the semi-supervised S3VM over the dataset. The last row gives the number of cases in which each method achieves the best performance.

From Tables 3 and 4, we can make several observations as follows,

- When 10 instances are labeled, S3VM performs worse than SVM over 2 datasets, while the performance of ACA-S3VM is better than SVM on all 12 datasets. When 100 instances are labeled, S3VM performs worse than SVM over 7 datasets, while the performance of ACA-S3VM is worse than SVM on just 4 datasets. As a result, ACA-S3VM can indeed improve the safety of semi-supervised classification.
- When 10 instances are labeled, ACA-S3VM performs better than S3VM over 7 datasets, and when 100 instances are labeled, ACA-S3VM also performs better than S3VM over 9 datasets. As a result, through adjusting the cluster assumption, ACA-S3VM can improve the performance of cluster-assumption-based semi-supervised classification learning.

Table 3 Performance comparison with 10 labeled instances

dataset	SVM	S3VM	S3VM_*us*	MR	S3VM_ACA
automobile	77.52 ± 12.19	80.50 ± 11.39	78.15 ± 8.13	78.69 ± 9.71	**80.81 ± 12.18**$^{*}*$
house	76.29 ± 1.39	75.14 ± 2.16	76.08 ± 2.43	74.68 ± 2.74	**77.72 ± 1.58**$^{*}*$
ionosphere	77.95 ± 10.72	**79.75 ± 19.43**	77.95 ± 10.72	80.04 ± 18.94	79.30 ± 16.30*
sonar	49.19 ± 3.56	49.72 ± 2.64	49.19 ± 3.56	50.00 ± 1.77	**54.75 ± 5.47**$^{*}*$
wdbc	87.66 ± 1.02	90.25 ± 1.60	**90.66 ± 1.02**	88.73 ± 0.71	90.25 ± 1.60*
german	50.77 ± 23.88	57.20 ± 21.47	55.13 ± 19.13	56.23 ± 18.72	**70.73 ± 11.43**$^{*}*$
isolet	89.28 ± 0.90	**97.33 ± 1.17**	92.32 ± 1.21	96.52 ± 3.12	**97.33 ± 1.17***
BCI	53.84 ± 1.69	52.85 ± 1.96	53.89 ± 2.31	51.03 ± 2.27	**54.87 ± 1.98**$^{*}*$
digit1	50.67 ± 1.23	51.69 ± 1.67	51.82 ± 1.11	51.76 ± 1.32	**52.56 ± 0.90**$^{*}*$
g241c	73.72 ± 2.39	75.29 ± 2.12	73.90 ± 2.41	**76.05 ± 1.89**	75.03 ± 2.22*
g241n	48.38 ± 0.33	57.76 ± 0.98	58.02 ± 0.68	**59.62 ± 0.67**	58.07 ± 0.94$^{*}*$
USPS	72.02 ± 1.21	74.80 ± 1.13	73.28 ± 1.08	**80.01 ± 1.87**	72.89 ± 2.21*
No. of win	0	2	1	3	**7**

Table 4 Performance comparison with 100 labeled instances

dataset	SVM	S3VM	S3VM_*us*	MR	S3VM_ACA
automobile	**92.39 ± 4.05**	88.73 ± 3.03	90.83 ± 3.21	88.15 ± 6.07	90.24 ± 2.25$_{*}$
house	90.78 ± 2.28	91.19 ± 0.01	90.86 ± 2.34	90.22 ± 0.61	**91.72 ± 0.01**$^{*}*$
ionosphere	**93.65 ± 0.48**	91.49 ± 0.36	93.43 ± 0.27	91.08 ± 0.62	92.86 ± 0.36$_{*}$
sonar	**85.56 ± 6.53**	80.83 ± 6.70	84.56 ± 6.53	80.05 ± 10.66	83.51 ± 7.01$_{*}$
wdbc	88.42 ± 1.47	87.28 ± 4.17	88.56 ± 2.32	88.31 ± 1.53	**89.20 ± 4.24**$^{*}*$
german	59.88 ± 9.19	**72.24 ± 8.66**	68.29 ± 7.35	72.02 ± 6.89	72.23 ± 7.62*
isolet	91.68 ± 1.97	98.08 ± 1.75	**98.67 ± 1.87**	98.27 ± 1.88	98.08 ± 1.75*
BCI	**55.92 ± 3.54**	53.74 ± 2.97	55.86 ± 3.12	52.96 ± 2.86	54.37 ± 3.01$_{*}$
digit1	51.00 ± 6.47	50.97 ± 8.40	51.05 ± 6.47	50.03 ± 7.21	**52.03 ± 4.46**$^{*}*$
g241c	76.89 ± 2.32	79.24 ± 2.17	78.32 ± 2.56	78.13 ± 2.38	**79.89 ± 1.94**$^{*}*$
g241n	51.89 ± 1.08	51.51 ± 1.36	51.67 ± 1.45	51.92 ± 1.41	**52.05 ± 0.57**$^{*}*$
USPS	90.23 ± 0.65	94.32 ± 0.43	93.20 ± 0.56	**95.32 ± 1.08**	93.79 ± 0.65*
No. of win	4	1	1	1	**5**

- When 10 instances are labeled, ACA-S3VM performs better than S3VM_*us* over 10 of the 12 datasets, when 100 instances are labeled, ACA-S3VM performs better than S3VM_*us* over 7 datasets. As a result, it is reasonable to consider the different confidences of individual unlabeled instances in semi-supervised classification.
- ACA-S3VM performs the best over 7 out of the 12 datasets when 10 instances are labeled, and performs the best over 5 datasets when 100 instances are labeled. It exactly shows the effectiveness of ACA-S3VM.
- It can also been found that when 10 instances are labeled, S3VM usually performs better than LS-SVM, while 100 instances are labeled, S3VM can performs worse than LS-SVM in some cases. The reason is that when the labeled instances are extremely limited, adopting the unlabeled instances can help improve the performance, while given enough

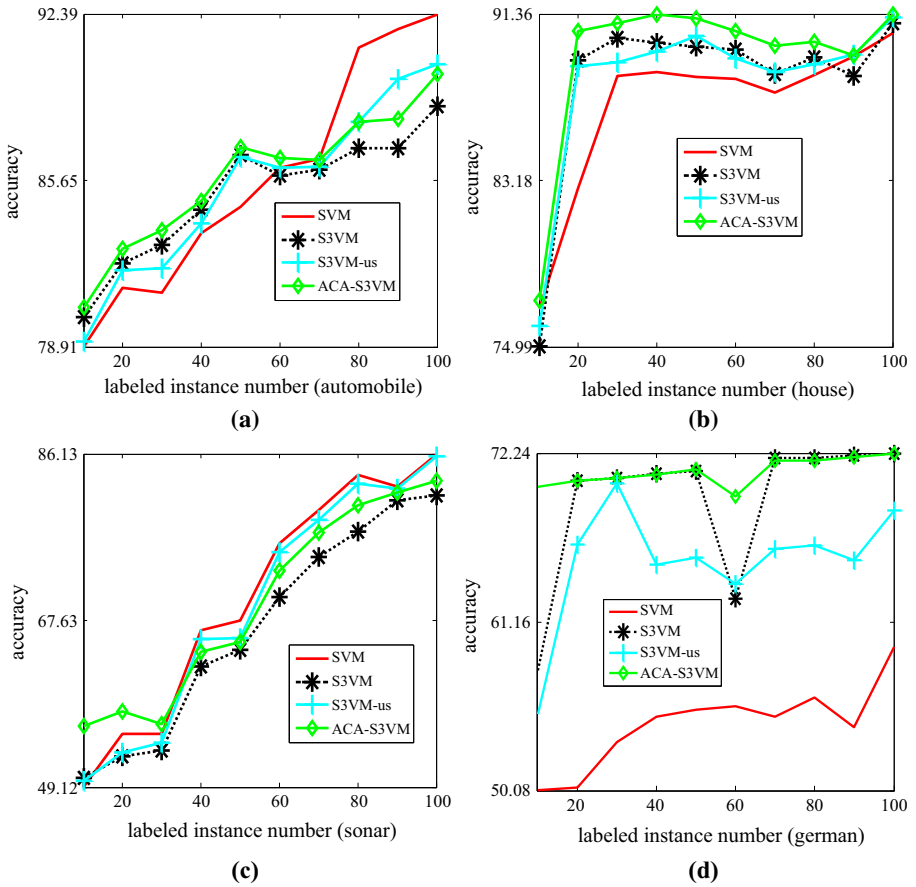

Fig. 2 The performances of the compared methods w.r.t different numbers of labeled instances from {10, 20, 30, 40, 50, 60, 70, 80, 90, 100}

instances labeled, using the unlabeled instance may not help learning, or even hurt the performance. As a result, it is necessary to study a safe usage of unlabeled instances, which is exactly what ACA-S3VM does.

4.2 Comparison with Different Number of Labeled Instances

We also show the performances of the compared methods with different numbers of labeled instances from {10, 20, 30, 40, 50, 60, 70, 80, 90, 100} in Fig. 2. From Fig. 2, we can find that,

(1) For dataset *automobile* in Fig. 2a, when the labeled instances is less than 60, S3VM performs better than SVM, in those cases, ACA-S3VM performs better than the other methods, thus it improves the performance of semi-supervised classification. When the number of labeled instances is larger than 60, S3VM performs worse than SVM, and so does S3VM_*us* and ACA_S3VM, because the given labeled instances may be enough to construct the ideal decision boundary. However, both S3VM_*us* and ACA_S3VM perform better than S3VM, as a result, they can both improve the safety of semi-supervised

classification. It is worth noting that semi-supervised classification makes sense when the labeled instances are extremely limited, in this case, ACA_S3VM can indeed provide a better performance for semi-supervised classification.

(2) For dataset *house* in Fig. 2b, S3VM performs better than SVM in most cases, and at the same time, ACA_S3VM performs better than both SVM and S3VM in all cases. In fact, ACA_S3VM performs the best in all cases, indicating its effectiveness for boosting semi-supervised classification.

(3) For dataset *sonar* in Fig. 2c, when the number of labeled instance is larger than 10, SVM performs better than S3VM in all cases, thus in those cases, adopting the unlabeled instances leads to unsafe learning. At the same time, both S3VM_*us* and ACA-S3VM can improve the safety to some extent. When the number of labeled instances is less than 40, ACA_S3VM performs better than SVM, actually it performs the best among the compared methods in this case. As a result, ACA_S3VM can improve the safety of semi-supervised classification. It is worth noting that when the labeled instances are extremely limited, the performance of ACA_S3VM is inspiring.

(4) For dataset *german* in Fig. 2d, ACA_S3VM performs the best in most cases, especially when the performances of S3VM are not desirable with 10 and 60 labeled instances, ACA_S3VM can achieve much better performance. As a result, ACA_S3VM can boost the performance of semi-supervised classification.

As a result, through adjusting the cluster assumption, ACA-S3VM can improve the safety of semi-supervised classification based on cluster assumtion, and further boost its performance. At the same time, it is easily found that the classification performance can degenerate with the increase of the labeled instances. Because the labeled instances are randomly selected, the labeled instances critical for classification can be selected, and at the same time, labeled instances not helpful for learning may also be select, which will mislead the classification.

5 Conclusion

Semi-supervised classification method may yield even worse performance than the corresponding supervised counterpart. It naturally reduces the confidence for applying semi-supervised methods to real applications, and thus it is desired to develop safe semi-supervised classification methods never performing worse than the supervised counterparts. Since one possible reason for such unsafe learning is that the cluster assumption may not well reflect the real data distribution, i.e., when samples from different classes are seriously overlapped, the real boundary actually lies not in the low density region, which can not be detected by the cluster assumption. However, an unsupervised clustering method is able to find the real boundary in this case. As a result, we develop a novel safe semi-supervised classification method ACA-S3VM in this paper based on the cluster assumption adjusted by clustering, in which instances in the cluster boundary will also be classified into the class boundary. In this way, the class boundary is guided to pass through the cluster boundary rather than the low-density region in the seriously overlapped case. Experiments over both UCI and benchmark datasets demonstrate the effectiveness of ACA_S3VM compared with the state-off-the-art semi-supervised classification methods. Besides, the idea proposed in this paper can be combined with other safe semi-supervised methods, such as SA-SSCCM and S4VM, to further improve the learning safety.

We demonstrate the effectiveness of ACA_S3VM by empirically comparison with other semi-supervised classification methods, however, some theoretical guarantees is still needed, for example, the analysis of the Rademacher complexity [23], which will be one of our important future work.

Acknowledgements This work was supported by the National Natural Science Foundation of China under Grant Nos. 61300165, 61375057 and 61300164, the Specialized Research Fund for the Doctoral Program of Higher Education of China under Grant No. 20133223120009, the Introduction of Talent Research Foundation of Nanjing University of Posts and Telecommunications under Grant Nos. NY213033 and NY213031.

References

1. Zhou Z-H, Li M (2010) Semi-supervised learning by disagreement. Knowl Inf Syst 24(3):415–439
2. Zhu X, Goldberg AB (2009) Introduction to semi-supervised learning. Morgan & Claypool, San Rafael
3. Zhu X (2008) Semi-supervised learning literature survey. University of Wisconsin-Madison, Computer Sciences, Madison
4. Chapelle O, Scholkopf B, Zien A (2006) Semi-supervised learning. MIT Press, Cambridge
5. Gong C et al (2015) Scalable semi-supervised classification via Neumann series. Neural Process Lett 42(1):187–197
6. Zhao Z-Q et al (2010) A modified semi-supervised learning algorithm on Laplacian eigenmaps. Neural Process Lett 32(1):75–82
7. Mallapragada PK et al (2009) Semiboost: boosting for semi-supervised learning. IEEE Trans Pattern Anal Mach Intell 31(11):2000–2014
8. Fung G, Mangasarian OL (2001) Semi-supervised support vector machine for unlabeled data classification. Opt Methods Softw 15(1):99–105
9. Collobert R et al (2006) Large scale transductive SVMs. J Mach Learn Res 7:1687–1712
10. Li Y-F, Kwok JT, Zhou Z-H (2009) Semi-supervised learning using label mean. In: Proceedings of the 26th international conference on machine learning. Montreal, Canada
11. Bengio Y, Alleau OB, Le Roux N (2006) Label propagation andquadratic criterion. In: Chapelle O, Schölkopf B, Zien A (eds) Semi-supervised learning. MIT Press, Cambridge, pp 193–216
12. Zhu X, Ghahramani Z (2002) Learning from labeled and unlabeled data with label propagation. Carnegie Mellon University, Pittsburgh
13. Belkin M, Niyogi P, Sindhwani V (2006) Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J Mach Learn Res 7(11):2399–2434
14. Li Y-F, Zhou Z-H (2011) Improving semi-supervised support vector machines through unlabeled instances selection. In: Proceedings of the 25th AAAI conference on artificial intelligence (AAAI'11). San Francisco, CA
15. Li Y-F, Zhou Z-H (2011) Towards making unlabeled data never hurt. In: Proceedings of the 28th international conference on machine learning (ICML'11). Bellevue, WA
16. Wang Y, Chen S (2013) Safety-aware semi-supervised classification. IEEE Trans Neural Netw Learn Syst 24(11):1763–1772
17. Li Y-F, Zhou Z-H (2015) Towards making unlabeled data never hurt. IEEE Trans Pattern Anal Mach Intell 37(1):175–188
18. Wang Y, Chen S, Zhou Z-H (2012) New semi-supervised classification method based on modified cluster assumption. IEEE Trans Neural Netw Learn Syst 23(5):689–702
19. Soares RGF, Chen H, Yao X (2012) Semi-supervised classification with cluster regularisation. IEEE Trans Neural Netw Learn Syst 23(11):1779–1792
20. Gu B, Sheng VS (2016) A robust regularization path algorithm for ν-support vector classification. IEEE Trans Neural Netw Learn Syst 1:1–8
21. Joachims T (1999) Transductive inference for text classification using support vector machines. In: Proceedings of the 16th international conference on machine learning. Bled, Slovenia
22. Gorski J, Pfeuffer F (2007) Biconvex sets and optimization with biconvex functions: a survey and extensions. Math Methods Oper Res 66(3):373–407
23. Anguita D et al (2014) Unlabeled patterns to tighten Rademacher complexity error bounds for kernel classifiers. Pattern Recognit Lett 37:210–219