

# Least Squares One-class Support Vector Machine on Fuzzy Set

Jingjing Zhang<sup>1</sup> and Ping Zhong<sup>1\*</sup>

<sup>1</sup>College of Science, China Agricultural University, 100083 Beijing, China  
zhjingjing09@163.com, \* Corresponding author: zping@cau.edu.cn

## Abstract

*In this paper, we formulate a least squares version of the one-class support vector fuzzy machine (LS one-class SVFM) which is combined with the fuzzy set theory. The parameters in the proposed algorithm, such as weight vector and bias term, are fuzzy numbers. Our model only needs to solve a system of linear equations, instead of a complex quadratic programming problem (QPP) solved in one-class SVFM. Our experiments on publicly available datasets indicate that our model has comparable classification accuracy to that of the one-class SVFM but with remarkably less computational time.*

**Keywords:** Support vector machine; Fuzzy system; Least squares; Class imbalanced; One-class SVM

## 1. Introduction

Learning from imbalanced or skewed datasets is an important issue in machine learning research. There are many class imbalanced problems in real-world applications [1,2,3]. For two-class imbalanced problems, one class is made up of a large number of samples while the other one is made up of only a few samples. The traditional algorithms are not suitable to deal with imbalanced dataset, since the separating hyperplane tends to the minority samples, which makes them misclassify easily. However, the minority is always the more important class. Improving the accuracy of minority class is a meaningful research.

To deal with the imbalanced dataset, there are probably two types of approaches. One type mainly contains under-sampling from majority class [4,5] or over-sampling from minority class [6,7]. The other type is to modify the learning algorithms such as adjusting the cost method [8,9], kernel-based method [10], margin calibration [11]. One-class support vector machine (one-class SVM) is widely used in the heavily class imbalanced problems, such as the context of information retrieval [12], hidden information [13], outlier detection [14] and so on. The one-class SVM aims to distinguish one class of samples from the rest. It constructs a maximal margin hyperplane to separate the training samples from the origin [15,16]. Many efforts have been made to reduce the time complexity or to raise the classification accuracy. The authors merged the rough set theory with one-class SVM to enhance the performance of the classifier [17]. In [18], the author proposed the least squares one-class SVM reducing the training time. In [19], the authors proposed the reduced least squares one-class SVM in empirical feature space.

In some systems, we consider a fuzzy structure when the available information is uncertain. The fuzzy structure is described by a fuzzy function whose parameters are given by fuzzy set [20]. Since the excellent of the SVM, many researches have been made apply the fuzzy set theory to SVM for classification and regression problems. In [21], the author gave the model of SVM with fuzzy set theory for two class problem. In [22], the authors incorporated the concept of fuzzy set theory into the support vector regression (SVR). In [23], the authors proposed the regularized LS SVR in fuzzy set theory.

Recently, Hao proposed the one-class SVM with fuzzy model, called one-class SVFM [24]. It constructs a fuzzy hyperplane to distinguish the target class from the rest. An effective combination of unseen samples and fuzzy set theory is useful to find a fuzzy structure in an evaluation system. However, the one-class SVFM needs to solve a time-consuming quadratic programming problem (QPP).

In this paper, we propose a least squares one-class support vector fuzzy machine, called LS one-class SVFM. Different from the one-class SVFM, the proposed algorithm introduces a quadratic loss function in objective function and equalities in the constraints. To solve the optimization problem of the proposed algorithm only requires solving a set of linear equations, instead of solving a complex QPP. Thus our algorithm owns low time complexity. Computational comparisons for linear kernel and nonlinear kernel between the proposed LS one-class SVFM and one-class SVFM on both generalization performance and training time have been made on several benchmark datasets. The experimental results show that the proposed method has a fast training speed, and it also acquires comparable accuracies with the one-class SVFM.

This paper is organized as follows. Section 2 briefly dwells on the one-class SVFM. Section 3 gives our least squares one-class SVFM. Experiments are presented in Section 4. Section 5 concludes this paper.

## 2. The One-Class SVFM

In this paper, the training samples are denoted by a set  $T = \{x_1, x_2, \dots, x_l\}$ , where  $x_i \in R^n$ .

The one-class SVFM is constructed based on one-class SVM and fuzzy theory [20]. It searches for a fuzzy hyperplane that separates the target class from the fuzzy origin. The parameters, such as the weight vector and bias term, are fuzzy numbers. For computational simplicity, the symmetric triangular fuzzy numbers are chosen [20]. The fuzzy hyperplane is described as

$$f(x) = W^T x + B, \text{ for } W \in T(R)^n, \quad B \in T(R) \quad (1)$$

where  $T(R)$  denotes the space of all symmetric triangular fuzzy numbers. The fuzzy weight vector  $W = (w, c)$  is described by center  $w$  and width  $c$ , while  $B = (b, d)$  is the fuzzy bias term, described by center  $b$  and width  $d$ . For a sample  $x$ ,  $f(x) = W^T x + B$  is a symmetric triangular fuzzy number with center  $w^T x + b$  and width  $c^T |x| + d$ , where  $|x| = (|x_1|, \dots, |x_n|)$ .

To find  $f(x)$ , the one-class SVFM considers the following QPP

$$\begin{aligned} \min_{w, c, b, d, \xi_i} & \frac{1}{2} \|w\|^2 + b + M \left( \frac{1}{2} \|c\|^2 + d \right) + C \sum_{i=1}^l \mu_i \xi_i \\ \text{s.t.} & \quad W^T x_i + B \underset{f}{\geq} \Theta - \xi_i, \quad i = 1, \dots, l \end{aligned} \quad (2)$$

where  $0 < \mu_i \leq 1$  is the fuzzy membership. The parameter  $C > 0$  is to balance the model complexity and the empirical risk. The symbol " $\underset{f}{\geq}$ " denotes the fuzzy larger than, and  $\Theta$  represents the fuzzy number with center zero and width  $O_w$ . The parameters  $0 < M, O_w \leq 1$  control the fuzzy range. Minimization the term  $\|w\|^2$  is equivalent to the maximization of the margin between the target class from the origin. The slack variable

$\xi_i$  represents the error caused by a training sample  $x_i$  with respect to the fuzzy hyperplane.

$\frac{1}{2}(\|c\|^2 + d)$  represents the vagueness of the model. The more vague of model, the more inaccurate results obtain. Problem (2) can be written as

$$\begin{aligned} \min_{w,c,b,c,\xi_i} & \frac{1}{2}\|w\|^2 + b + M\left(\frac{1}{2}\|c\|^2 + d\right) + C\sum_{i=1}^l \mu_i(\xi_{1i} + \xi_{2i}) \\ \text{s.t.} & (w^T x_i + b) + (c^T |x_i| + d) \geq O_w - \xi_{1i}, \\ & (w^T x_i + b) - (c^T |x_i| + d) \geq -O_w - \xi_{2i}, \\ & d \geq 0, \xi_{1i}, \xi_{2i} \geq 0, \quad i = 1, \dots, l \end{aligned} \quad (3)$$

where  $|\cdot|$  is to ensure the width hyperplane  $c^T |x_i| + d > 0$ .

The dual problem (2) can be formulated as follows by KKT conditions.

$$\begin{aligned} \max_{\alpha_{1i}, \alpha_{2i}} & -\frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l (\alpha_{1i} + \alpha_{2i})(\alpha_{1j} + \alpha_{2j}) x_i^T x_j \\ & - \frac{1}{2M} \sum_{i=1}^l \sum_{j=1}^l (\alpha_{1i} - \alpha_{2i})(\alpha_{1j} - \alpha_{2j}) |x_i|^T |x_j| + \sum_{i=1}^l (\alpha_{1i} - \alpha_{2i}) O_w \\ \text{s.t.} & \sum_{i=1}^l (\alpha_{1i} + \alpha_{2i}) = 1 \\ & \sum_{i=1}^l (\alpha_{1i} - \alpha_{2i}) \leq M \\ & 0 \leq \alpha_{1i} \leq C\mu_i, 0 \leq \alpha_{2i} \leq C\mu_i, i = 1, \dots, l \end{aligned} \quad (4)$$

Once the solution of  $(\alpha_{1i}, \alpha_{2i}, \dots, \alpha_{1l}, \alpha_{2l})^T$  is obtained, the weight vectors  $w$  and  $c$  can be obtained as follows.

$$w = \sum_{i=1}^l (\alpha_{1i} + \alpha_{2i}) x_i, \quad c = \frac{1}{M} \sum_{i=1}^l (\alpha_{1i} - \alpha_{2i}) |x_i|.$$

For some  $\alpha_i \in (0, C\mu_i)$  and  $\alpha_j \in (0, C\mu_j)$ , according to the KKT conditions, we have bias terms  $b$  and  $d$ .

$$\begin{aligned} b &= -\frac{1}{2} (w^T x_i + w^T x_j + c^T |x_i| - c^T |x_j|) \\ d &= -\frac{1}{2} (w^T x_i - w^T x_j + c^T |x_i| + c^T |x_j| - 2O_w). \end{aligned}$$

For a new sample  $x_i$ , we evaluate it falls on which side of the hyperplane by defining the fuzzy partial ordering relation. Given two symmetric triangular fuzzy numbers  $P = (m_p, m_p)$  and  $Q = (m_q, m_q)$ , we have that

$$R_{\geq Q} P = R(P, Q) = \begin{cases} 1 & \text{if } A > 0 \ \& \ B > 0, \\ 0 & \text{if } A < 0 \ \& \ B < 0, \\ 0.5(1 + (A + B) / \max(|A|, |B|)) & \text{if } o.w. \end{cases} \quad (5)$$

where  $A = (m_p + c_p) - (m_q + c_q)$  and  $B = (m_p - c_p) - (m_q - c_q)$ .

For (5), we observe that

$$R(P, Q) = \begin{cases} > 0.5 & \text{if } m_p > m_Q, \\ = 0.5 & \text{if } m_p = m_Q, \\ < 0.5 & \text{if } m_p < m_Q. \end{cases} \quad (6)$$

Hence, the decision function of the one-class SVFM can be expressed as

$$f(x_i) = R_{\geq \Theta} (W^T x_i + B) = R(W^T x_i + B, \Theta). \quad (7)$$

### 3. Least Squares One-Class Support Vector Fuzzy Machine

#### 3.1. Linear Kernel Case

Inspired by the one-class SVFM, we propose a least squares one-class SVFM. The LS one-class SVFM can be derived by solving the following optimization problem

$$\begin{aligned} \min_{w, c, b, c, \xi_i} & \frac{1}{2} \|w\|^2 + b + M \left( \frac{1}{2} \|c\|^2 + d^2 \right) + \frac{C}{2} \sum_{i=1}^l \mu_i \xi_i^2 \\ \text{s.t.} & W^T x_i + B \underset{f}{=} \Theta - \xi_i, \quad i = 1, \dots, l \end{aligned} \quad (8)$$

where  $0 < \mu_i \leq 1$  is the fuzzy membership. The parameter  $C > 0$  is the regularization parameter which determines the penalty weight, and  $\xi_i$  is the slack vector. The parameter  $0 < M \leq 1$  controls the fuzzy origin. The symbol " $\underset{f}{=}$ " represents the fuzzy equal.

We rewrite the primal problem (8) as

$$\begin{aligned} \min_{w, c, b, c, \xi_i} & \frac{1}{2} \|w\|^2 + b + M \left( \frac{1}{2} \|c\|^2 + d^2 \right) + \frac{C}{2} \sum_{i=1}^l \mu_i (\xi_{1i}^2 + \xi_{2i}^2) \\ \text{s.t.} & (w^T x_i + b) + (c^T |x_i| + d) = O_w - \xi_{1i}, \quad i = 1, \dots, l \\ & (w^T x_i + b) - (c^T |x_i| + d) = -O_w - \xi_{2i}, \quad i = 1, \dots, l \end{aligned} \quad (9)$$

In problem (9), the first term  $\frac{1}{2} \|w\|^2$  in the objective function is to control the model complexity for finding the fuzzy hyperplane.  $\frac{1}{2} (\|c\|^2 + d^2)$  is the term which characterizes the vagueness of the model. The slack variables  $\xi_{1i}, \xi_{2i}$  represent the loss caused by the fuzzy hyperplane to the fuzzy region.  $0 < O_w \leq 1$  represents width of fuzzy region  $\Theta$ . Different from (3), we reformulate our method by using a quadratic loss function and equality constraints. In order to solve problem (9), we construct its Lagrangian function

$$\begin{aligned} L = & \frac{1}{2} \|w\|^2 + b + M \left( \frac{1}{2} \|c\|^2 + d^2 \right) + \frac{C}{2} \sum_{i=1}^l \mu_i (\xi_{1i}^2 + \xi_{2i}^2) \\ & - \sum_{i=1}^l \alpha_i (w^T x_i + b + c^T |x_i| + d - O_w + \xi_{1i}) \\ & - \sum_{i=1}^l \beta_i (w^T x_i + b - c^T |x_i| - d + O_w + \xi_{2i}) \end{aligned} \quad (10)$$

where  $\alpha = (\alpha_1, \dots, \alpha_l)^T$  and  $\beta = (\beta_1, \dots, \beta_l)^T$  are the Lagrangian multiplier vectors. Differentiating  $L$  with respect to  $w, c, b, d, \xi_{1i}, \xi_{2i}, \alpha_i, \beta_i$  and setting the results to zero, we obtain

$$\partial L / \partial w = w - \sum_{i=1}^l (\alpha_i + \beta_i) x_i = 0 \Rightarrow w = \sum_{i=1}^l (\alpha_i + \beta_i) x_i \quad (11)$$

$$\partial L / \partial c = M c - \sum_{i=1}^l \alpha_i |x_i| + \sum_{i=1}^l \beta_i |x_i| = 0 \Rightarrow c = \frac{1}{M} \sum_{i=1}^l (\alpha_i - \beta_i) |x_i| \quad (12)$$

$$\partial L / \partial b = 1 - \sum_{i=1}^l \alpha_i - \sum_{i=1}^l \beta_i = 0 \Rightarrow \sum_{i=1}^l (\alpha_i + \beta_i) = 1 \quad (13)$$

$$\partial L / \partial d = M d - \sum_{i=1}^l \alpha_i + \sum_{i=1}^l \beta_i = 0 \Rightarrow d = \frac{1}{M} \sum_{i=1}^l (\alpha_i - \beta_i) \quad (14)$$

$$\partial L / \partial \xi_{1i} = C \mu_i \xi_{1i} - \alpha_i = 0 \Rightarrow \xi_{1i} = \frac{\alpha_i}{C \mu_i} \quad (15)$$

$$\partial L / \partial \xi_{2i} = C \mu_i \xi_{2i} - \beta_i = 0 \Rightarrow \xi_{2i} = \frac{\beta_i}{C \mu_i} \quad (16)$$

$$\partial L / \partial \alpha_i = w^T x_i + b + c^T |x_i| + d - O_w + \xi_{1i} = 0 \quad (17)$$

$$\partial L / \partial \beta_i = w^T x_i + b - c^T |x_i| - d + O_w + \xi_{2i} = 0 \quad (18)$$

From the above conditions for optimality,  $b$ ,  $\alpha$  and  $\beta$  can be solved by following linear equations

$$\begin{pmatrix} 0 & I & I \\ I^T & K + \frac{K_1}{M} + \frac{E}{M} + V & K - \frac{K_1}{M} - \frac{E}{M} \\ I^T & K - \frac{K_1}{M} - \frac{E}{M} & K + \frac{K_1}{M} + \frac{E}{M} + V \end{pmatrix} \begin{pmatrix} b \\ \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} 1 \\ -O_w \\ O_w \end{pmatrix} \quad (19)$$

where  $K = (K_{ij})_{l \times l}$  is a matrix with  $K_{ij} = x_i^T x_j$ , and  $K1 = (K1_{ij})_{l \times l}$  is also a matrix with  $K1_{ij} = |x_i|^T |x_j|$ .  $\alpha$  and  $\beta$  denote the column vectors of Lagrangian multipliers.  $I$  represents the all one row vector of  $n$  dimension.  $E$  denotes the identity matrix.  $V = \text{diag}\{\frac{1}{C \mu_1}, \dots, \frac{1}{C \mu_l}\}$  is a diagonal matrix. Utilizing the solution of (19), the optimal hyperplane of our algorithm can be expressed as

$$f(x) = W^T x + B \quad (20)$$

where  $W = (\sum_{i=1}^l (\alpha_i + \beta_i) x_i, \frac{1}{M} \sum_{i=1}^l (\alpha_i - \beta_i) |x_i|)$  and  $B = (b, \frac{1}{M} \sum_{i=1}^l (\alpha_i - \beta_i))$ .

### 3.2. Nonlinear Kernel Case

In order to extend our results to the nonlinear case, we consider the kernel-generated fuzzy hyperplane as

$$f(x) = W^T \varphi(x) + B \quad (21)$$

where  $\varphi(\cdot)$  is to map a dataset  $\chi$  to high dimensional feature space  $H$ , and  $k(\cdot, \cdot)$  is the kernel function. Similar to the linear case, the nonlinear LS one-class SVFM can be derived through solving the following optimization problem

$$\begin{aligned} \min_{w,c,b,c,\xi_i} & \frac{1}{2} \|w\|^2 + b + M \left( \frac{1}{2} \|c\|^2 + d^2 \right) + \frac{C}{2} \sum_{i=1}^l \mu_i (\xi_{1i}^2 + \xi_{2i}^2) \\ \text{s.t.} & \quad (w^T \varphi(x_i) + b) + (c^T \varphi(|x_i|) + d) = O_w - \xi_{1i}, \quad i = 1, \dots, l \\ & \quad (w^T \varphi(x_i) + b) - (c^T \varphi(|x_i|) + d) = -O_w - \xi_{2i}, \quad i = 1, \dots, l \end{aligned} \quad (22)$$

Similar with the linear case, the solution of problem (22) is

$$\begin{pmatrix} 0 & I & I \\ I^T & K + \frac{K_1}{M} + \frac{E}{M} + V & K - \frac{K_1}{M} - \frac{E}{M} \\ I^T & K - \frac{K_1}{M} - \frac{E}{M} & K + \frac{K_1}{M} + \frac{E}{M} + V \end{pmatrix} \begin{pmatrix} b \\ \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} 1 \\ -O_w \\ O_w \end{pmatrix} \quad (23)$$

where  $I = (1, 1, \dots, 1)$ ,  $V = \text{diag}\{\frac{1}{C\mu_1}, \dots, \frac{1}{C\mu_l}\}$ , and  $E$  denotes the identity matrix.

$K = (K_{ij})_{l \times l}$  is the kernel matrix with  $K_{ij} = \varphi(x_i)^T \varphi(x_j)$ , and  $K1 = (K1_{ij})_{l \times l}$  is also the kernel matrix with  $K1_{ij} = \varphi(|x_i|)^T \varphi(|x_j|)$ .

Utilizing the solution of (23), the optimal hyperplane of nonlinear LS one-class SVFM is expressed as

$$f(x) = W^T \varphi(x) + B \quad (24)$$

where  $W = (\sum_{i=1}^l (\alpha_i + \beta_i) \varphi(x_i), \frac{1}{M} \sum_{i=1}^l (\alpha_i - \beta_i) \varphi(|x_i|))$  and  $B = (b, \frac{1}{M} \sum_{i=1}^l (\alpha_i - \beta_i))$ .

## 4. Experiments

To check the validity of the proposed LS one-class SVFM in dealing with the class imbalanced problem, we compare it with the one-class SVFM on several benchmark datasets [25]. Both the algorithms are implemented on MATLAB 2012a under Windows 7 running on a PC. We utilize “quadprog.m” for one-class SVFM, “inv.m” for LS one-class SVFM. The parameters in the linear case,  $(C, O_w, M)$  are employed. In the nonlinear case, gaussian kernel function  $K(x, y) = \exp(-p \|x - y\|^2)$  is considered for all datasets, and the parameters  $(p, C, O_w, M)$  are employed. We select the values of parameters

$$O_w, M \in \{0.1, 0.2, \dots, 1\}$$

and  $(p, C) \in \{2^i \mid i = -10, -9, \dots, 10\} \times \{10^i \mid i = -4, -3, \dots, 5\}$ . To search for the optimal parameters, we derive  $(C, O_w, M)$  linear case and  $(p, C, O_w, M)$  for nonlinear case on each training dataset by cross-validation approach. For LS one-class SVFM, since the optimal hyperplane is a distance function, we discriminate a test sample  $x$  as positive (*i.e.*, it belongs to the target class) if it satisfies  $\varepsilon^- \leq f(x) \leq \varepsilon^+$ , where  $\varepsilon^+ = \max_k f(x_k)$ ,  $\varepsilon^- = \min_k f(x_k)$  with  $x_k$  being a training sample.

**Table 1. Description of Seven Data Sets**

Data set	Minority	Majority	feature	Data size
Glass	29	185	9	214
Wine	59	119	13	178
Vehicle	168	428	18	596
Breast Cancer	239	444	10	683
Segment	323	1879	18	2202
Abalone	391	3786	7	4177
fourclass	305	550	2	855

We evaluate the performance of LS one-class SVFM and one-class SVFM on seven UCI datasets, which consist of Glass, Wine, Vehicle, Breast Cancer, Segment, Abalone, and fourclass. In these datasets, Vehicle, Breast Cancer and fourclass are two-class imbalanced datasets. The minority of the dataset is regarded as the positive class, and the majority one is regarded as the negative class. The others are multi-class datasets. For Glass, Wine, Segment and Abalone, we use the 6th, 1st, 1st and 7th class as the positive class, respectively. Table 1 shows the detail descriptions of these datasets. Here several performance criteria based on the confusion matrix are utilized to give a more comprehensive and objective assessment. The confusion matrix consists of true positive (TP), false positive (FP), false negative (FN) and true negative (TN). To evaluate the performance of the algorithms on imbalanced datasets, we take sensitivity, specificity, G-mean, AUC and training time as performance criteria as follows.

$$(i) \text{ sensitivity} = \frac{TP}{TP + FN}$$

$$(ii) \text{ specificity} = \frac{TN}{TN + FP}$$

$$(iii) \text{ G-mean} = \sqrt{\text{sensitivity} \times \text{specificity}}$$

Sensitivity and specificity are defined as the accuracies on the positive class and the negative class, respectively. The value of G-mean is high when both sensitivity and specificity values are high. Higher value of G-mean implies that the algorithm has better performance. The area under the curve of receiver operating characteristics defined as AUC. It is used to measure the algorithm performance in a quantitative manner. The larger AUC, the better the algorithm performs. Note that all the samples are normalized as in [0,1] before learning.

We give the criterion to associate the fuzzy membership to each sample [20,26]. Given the training samples  $(x_1, \dots, x_l)$ . Let the mean of target class as  $x_{mean}$ , and the radius of the target class is defined as  $r_{target} = \max_i |x_i - x_{mean}|, i = 1, \dots, l$ . The fuzzy membership

$$\mu_i \text{ is defined as } \mu_i = 1 - \frac{|x_i - x_{mean}|}{(r_{target} + \delta)}, \text{ where } \delta > 0 \text{ is used to avoid } \mu_i = 0.$$

**Table 2. Results of Linear Classifiers on Benchmark Datasets. I and II Stand for One-Class SVFM and LS One-Class SVFM Respectively**

Data set	Alg	sen	spe	G-mean	AUC	Time(s)
Glass	I	0.5714	0.6605	0.6143	0.6286	0.0130
	II	<b>0.6667</b>	<b>0.6995</b>	<b>0.6829</b>	<b>0.6831</b>	<b>0.0001</b>
Wine	I	<b>0.8494</b>	0.4790	0.6378	0.6642	0.0671
	II	0.7714	<b>0.7613</b>	<b>0.7664</b>	<b>0.7577</b>	<b>0.0002</b>
Vehicle	I	<b>0.9909</b>	0.3458	<b>0.5854</b>	0.6684	0.3017
	II	0.9644	<b>0.5075</b>	0.4082	<b>0.7359</b>	<b>0.0008</b>
Breast Cancer	I	0.8640	0.9851	0.9226	0.9245	1.6106
	II	<b>0.8940</b>	<b>1.0000</b>	<b>0.9455</b>	<b>0.9470</b>	<b>0.0005</b>
Segment	I	0.8865	0.3035	0.5187	0.6034	1.6257
	II	<b>0.9815</b>	<b>0.6621</b>	<b>0.8061</b>	<b>0.8218</b>	<b>0.0019</b>
Abalone	I	0.7558	0.4660	0.5935	0.6109	3.7459
	II	<b>0.8100</b>	<b>0.8994</b>	<b>0.7727</b>	<b>0.8547</b>	<b>0.00009</b>
fourclass	I	0.5588	<b>0.7055</b>	0.6279	0.6321	2.9559
	II	<b>0.9804</b>	0.3824	<b>0.6389</b>	<b>0.6984</b>	<b>0.00002</b>

**Table 3. Results of Nonlinear Classifiers on Benchmark Datasets. I and II Stand for One-Class SVFM and LS One-Class SVFM Respectively**

Data set	Alg	sen	spe	G-mean	AUC	Time(s)
Glass	I	<b>0.6381</b>	0.9676	<b>0.7857</b>	<b>0.8028</b>	0.0051
	II	0.5143	<b>0.9914</b>	0.7140	0.7528	<b>0.0001</b>
Wine	I	0.7195	<b>1.0000</b>	<b>0.8110</b>	<b>0.8058</b>	0.0055
	II	<b>0.8961</b>	0.4553	0.6246	0.6657	<b>0.0001</b>
Vehicle	I	<b>0.9470</b>	0.8042	0.8727	0.8756	0.0171
	II	0.8932	<b>0.9416</b>	<b>0.9171</b>	<b>0.9174</b>	<b>0.0010</b>
Breast Cancer	I	0.9148	<b>0.9577</b>	<b>0.9360</b>	<b>0.9362</b>	0.2814
	II	<b>0.9742</b>	0.6743	0.8105	0.8243	<b>0.0024</b>
Segment	I	<b>0.9674</b>	0.9989	<b>0.9831</b>	<b>0.9832</b>	1.6257
	II	0.9395	<b>0.9993</b>	0.9689	0.9694	<b>0.0040</b>
Abalone	I	<b>0.8731</b>	0.6908	<b>0.7766</b>	0.7819	2.9833
	II	0.6600	<b>0.9047</b>	0.7727	<b>0.7824</b>	<b>0.00009</b>



	I	<b>0.9760</b>	0.8567	0.9144	0.9164	2.0379
fourclass	II	0.9517	<b>0.8902</b>	<b>0.9204</b>	<b>0.9210</b>	<b>0.0038</b>

Table 2 and Table 3 report the comparison results of these algorithms with linear and nonlinear kernels, respectively. They mainly contain the sensitivity (sen), specificity (spe), G-mean, the AUC and the training time (trT) on seven datasets. In Table 2, it has been seen that for the sensitivity index, the proposed method has better values than those of one-class SVFM on five datasets except Wine and Vehicle datasets. It indicates that the accuracy of positive class performs better in the proposed algorithm on these datasets. For the specificity, our method performs better on six of seven datasets. For G-mean, our method has better values on six datasets except Vehicle dataset. As for AUC, our method performs better on all datasets. Concerned with nonlinear one-class SVFM and nonlinear LS one-class SVFM, we observe that the values of sensitivity in our method perform better on Wine and Breast Cancer datasets, and it has comparable values with one-class SVFM on Segment and fourclass datasets. For specificity index, our algorithm owns bigger values on five of the seven datasets. As for G-mean, our algorithm has comparable values on five datasets except Wine and Breast Cancer datasets. For the evaluation criteria AUC, our method has comparable values on Segment, Ablaone and fourclass datasets, and has satisfactory results on Vehicle dataset. The training CPU time shown in Table 2 and Table 3 indicate that the proposed LS one-class SVFM is almost dozens of times faster than the one-class SVFM, since the algorithm solves a system of linear equations instead of a complex QPP in the one-class SVFM. In short, the proposed algorithm has comparable generalization performance with the one-class SVFM, and the training time is reduced greatly.

## 5. Conclusions

In this paper we propose a novel least squares (LS) one-class SVFM to deal with imbalanced problem. The proposed LS one-class SVFM incorporates the concept of fuzzy set theory and the least squares technology into the one-class SVM. Different from solving a dual QPP in one-class SVFM, we only need to solve a system of linear equations in LS one-class SVFM. It makes the LS one-class SVFM a fast speed while the one-class SVFM needs a high training time. Computational results on several UCI benchmark datasets demonstrate that LS one-class SVFM obtains the classification accuracy comparable to that of one-class SVFM algorithm with a fast learning speed.

## Acknowledgment

The work is supported by national Natural Science Foundation of China Grant No.11171346 and Chinese Universities Scientific Fund No.2013YJ010.

## References

- [1] D. Lewis and J. Catlett, "Training text classifiers by uncertainty sampling", In Proceedings of the Seventh Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland, (1994).
- [2] M. Kubat and S. Matwin, "Learning when negative examples abound", In Proceedings of the Ninth European Conference on Machine Learning, (1997), pp. 146-153.
- [3] P. Chan and S. Stolfo, "Toward scalable learning with nonuniform class and cost distributions: a case study in credit card fraud detection", In Proceedings of Knowledge Discovery and Data Mining, (1998), pp. 164-168.
- [4] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: one-sided selection", Proceedings of the 14th International Conference on Machine Learning, (1997), pp. 179-186.

- [5] G.E. Batista, R.C. Prati and M.C. Monard, "A study of the behavior of several methods for balancing machine learning training data", *ACM Sigkdd Explorations Newsletter*, vol. 6, (2004), pp. 20-29.
- [6] R. Barandela, R.M. Valdovinos, J.S. Sanchez and F.J. Ferri, "The imbalanced training sample problem: Under or over sampling?", In *Joint IAPR International Workshops on Structural, Syntactic, and Statistical Pattern Recognition, Lecture Notes in Computer Science*, vol. 3138, (2004), pp. 806-814.
- [7] N. Chawla, K. Bowyer, L. Hall and W. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling technique", *Journal of Artificial Intelligence Research*, vol. 16, (2002), pp. 321-357.
- [8] Y.H. Shao, W. J. Chen, J. J. Zhang, Z. Wang and N.Y. Deng, "An efficient weighted Lagrangian twin support vector machine for imbalanced data classification", *Pattern Recognition*, vol. 47, (2014), pp. 3158-3167.
- [9] K.N. Wang and P. Zhong, "Robust non-convex least squares loss function for regression with outliers", vol.71, (2014), pp. 290-302.
- [10] L. Ding, J. Watada and L.C. Chew, "A SVM-RBF method for solving imbalanced data problem", *ICIC Express Letters*, vol. 4, (2010), pp. 2419-2424.
- [11] Z. Zhao, P. Zhong and Y. Zhao, "Learning SVM with weighted maximum margin criterion for classification of imbalanced data", *Mathematical and Computer Modelling*, vol. 54, (2011), pp. 1093-1099.
- [12] L.M. Manevitz and M. Yousef. "One-class svms for document classification", *Journal of Machine Learning Research*, vol. 2, (2001), pp. 139-154.
- [13] W.X. Zhu and P. Zhong, "A new one-class SVM based on hidden information", *Knowledge-Based Systems*, vol. 60, (2014), pp. 35-43.
- [14] B. Chen, S.C. Chen and Z. S. Pan, "Survey of outlier detection technologies", *Journal of Shandong university*, (2009), vol. 39, pp. 13-23.
- [15] B. Schölkopf, J. Platt, J. Shawe-Taylor, A. Smola and R. Williamson, "Estimating the support of a high-dimensional distribution", *Neural Computation*, vol. 13, no. 7, (2001), pp. 1443 -1471.
- [16] D.M.J. Tax and R.P.W. Duin, "Support vector domain description", *Pattern Recognition Letters*, vol. 20, (1999), pp. 1191-1199.
- [17] Y. Xu and C. Liu, "A rough margin-based one class support vector machine", *Neural Comput & Applic*, vol. 22, (2013), pp. 1077-1084..
- [18] Y. Choi, "Least squares one-class support vector machine", *Pattern Recognition Letters*, vol. 30, (2009), pp. 1236-1240.
- [19] Z. Zhao, P. Zhong and Y. Zhao, "Reduced least squares one-class SVM in empirical feature space for imbalanced data", *ICIC Express Letters*, vol. 5, (2010), pp. 4115-4121.
- [20] D. Dubois and H. Prade, "Theory and applications, Fuzzy Sets and Systems", Academic Press, New York, (1980).
- [21] P.Y. Hao, "A fuzzy model of support vector classification algorithm", *IEEE International Conference on Fuzzy Systems*, (2008), pp. 299-304.
- [22] P.Y. Hao and J.H. Chiang, "Fuzzy regression analysis by support vector learning approach", *IEEE transactions on Fuzzy Systems*, vol. 16, (2008).
- [23] Y. Chen and Z. Zhong, "Regularized least squares support vector fuzzy regression", *Information Management and Engineering (ICIME)*, 2010 The 2nd IEEE International Conference, (2010), pp. 176-180.
- [24] P.Y. Hao, "Fuzzy one-class support vector machines", *Fuzzy Sets and Systems*, vol. 159, (2008), pp. 2317-2336.
- [25] C.L. Blake and C.J. Merz, "UCI repository for machine learning databases", Dept. of Information and Computer Sciences, Univ. of California, Irvine, (1998), <http://www.ics.uci.edu/~mlearn/MLRepository.html>.
- [26] C.F. Lin and S.D. Wang, "Fuzzy support vector machines", *IEEE Trans. Neural Networks*, vol. 13(2), (2002), pp. 464-471.

## Authors



**Jingjing Zhang**, he has received her B.S. degree in College of Mathematics and Information Science from Shaanxi Normal University, 2010, and received Master's degree in College of Science from China Agricultural University, China, in 2012. Currently, she is currently pursuing the Ph.D. degree at the College of Science of China Agricultural University. Her research interests include optimization methods ,machine learning and data mining.



**Ping Zhong**, he has received the B.S degree and Master degree in mathematics in 1994 and 1997 both from Qufu Normal University, China, and her Ph. D degree in 2002 from China Agricultural University. Now she is a professor of China Agricultural University. She has published more than 30 papers cited by SCI and EI. Her research interest includes optimization theory, machine learning and support vector machines.

