J Intell Inf Syst (2010) 35:333–345
DOI 10.1007/s10844-009-0102-y

Support vector classifier based on fuzzy *c*-means and Mahalanobis distance

Yong Zhang · Fuding Xie · Dan Huang · Min Ji

Received: 29 October 2007 / Revised: 11 August 2009 / Accepted: 11 August 2009 /
Published online: 21 August 2009

Abstract This paper presents a fuzzy support vector classifier by integrating modified fuzzy *c*-means clustering based on Mahalanobis distance into fuzzy support vector data description. The proposed algorithm can be used to deal with the outlier sensitivity problem in traditional multi-class classification problems. The modified fuzzy *c*-means clustering algorithm based on Mahalanobis distance takes into the samples' correlation account, and is improved to generate different weight values for main training data points and outliers according to their relative importance in the training data. Experimental results show that the proposed method can reduce the effect of outliers and give high classification accuracy.

Keywords Support vector data description · Fuzzy *c*-means algorithm · Mahalanobis distance · Support vector machine

1 Introduction

Founded on Vapnik's statistical learning theory, support vector machines (SVMs) (Vapnik 1995) have played an important role in many areas including pattern recognition, regression, image processing, and bioinformatics. However, unclassifiable regions exist when they are extended to multi-class problems. Now there are some methods to solve this problem. One is to construct several two-class classifiers, and then to assemble a multi-class classifier, such as one-against-one, one-against-all and DAGSVMs (Platt et al. 2002; Hsu and Lin 2002). The second method is to construct

Y. Zhang (✉) · F. Xie · D. Huang · M. Ji
Department of Computer, Liaoning Normal University,
No. 1, Liushu South Street, Ganjingzi, Dalian, Liaoning Province 116081, China
e-mail: ayong_zh@163.com

multi-class classifier directly, such as k-SVM proposed by Weston and Watkins (1998). Zhang et al. (2007) also proposed a fuzzy compensation multi classification SVM based on Weston and Walkins' idea. Recently, Zhu et al. proposed a multi-class classification algorithm which adopted the minimum enclosing spheres to classify a new example and showed that the resulting classifier performed comparable to the standard SVMs (Zhu et al. 2003). Based on Zhu et al. (2003), Wang et al. (2007) and Lee and Lee (2007) also proposed a new classification rule on the basis of Bayesian optimal decision theory respectively.

On the other hand, the obtained training data is often contaminated by noises in many practical engineering applications. Furthermore, some points in the training data set are misplaced far away from main body or even on the wrong side in feature space. The standard SVM is very sensitive to outliers and noises. Some techniques have been found to tackle this problem for SVM in the literature of data classification. One of the methods is to use averaging algorithm to relax the effect of outliers in Song et al. (2002); Hu and Song (2004). But in this kind of method, the additional parameter is difficult to be tuned. The second method, proposed in Lin and Wang (2002, 2003); Inoue and Abe (2001) and named as a fuzzy SVM (FSVM), is to apply fuzzy membership to the training data to relax the effect of the outliers. However, the selection of membership function remains a problem for FSVM so far. The third interesting method, called support vector data description (SVDD) (Tax and Duin 1999, 2004), has been shown effectiveness to detect outliers from the normal data points. However, it is well-known that SVDD is particular in the case of one-class classification problem.

The most common clustering method, the so-called fuzzy c-means clustering method (FCM) (Dunn 1974; Bezdek 1980), is based on the minimization of the distance-based objective function as (1). With fuzzy clustering it is not necessary to definitely place an object within one cluster, since the membership value of this object can be allocated among different clusters. This "distribution" of the membership among different clusters can be interpreted as the measure of similarity between a particular object and the respective clusters. However, traditional Euclidean distance is very difficult to depict the samples' correlation.

Clustering-based SVM (CB-SVM) (Yu et al. 2003) is a newly developed method to consider the clustering information in reducing the training samples for SVMs. However, the clustering is performed in the input space for nonlinear CB-SVM, while the distance is measured in the kernel space, which introduces unexpected mistakes in sample reduction.

In this paper we propose a fuzzy support vector classifier by integrating FCM method based on Mahalanobis distance into fuzzy support vector data description. This paper first presents a FCM method based on Mahalanobis distance, then gives a fuzzy support vector data description, which uses fuzzy membership degree of the cluster computed by the FCM method to partition training data. Accordingly, this paper proposes the multi classification algorithm based on the fuzzy support vector classifier, and give the classification rule.

The rest of this paper is organized as follows. In Section 2 we give a brief description of fuzzy c-means algorithm and Mahalanobis distance, and presents a FCM algorithm based on Mahalanobis distance. In Section 3 we present fuzzy support vector data description, and present the proposed fuzzy support vector classifier algorithm. In Section 4 we illustrate our proposed algorithm by comparing its performance with other methods. Finally, the conclusion is given in Section 5.

2 FCM algorithm and Mahalanobis distances

Fuzzy *c*-means (FCM) algorithm, has been shown to have satisfied ability of handling noises and outliers. In this section, we first introduce the fuzzy *c*-means algorithm, and review the distance measure based on Mahalanobis distance. We present the improved FCM algorithm based on Mahalanobis distance.

2.1 FCM algorithm

Clustering methods seeks to organize a set of items into clusters such that items within a given cluster have a high degree of similarity, whereas items belonging to different clusters have a high degree of dissimilarity.

However, conventional hard clustering methods restrict each point of the data set to exactly one cluster. Fuzzy clustering generates a fuzzy partition based on the idea of partial membership expressed by the degree of membership of each pattern in a given cluster. Dunn (1974) presented one of the first fuzzy clustering methods based on an adequacy criterion defined by the Euclidean distance. Bezdek (1980) further generalized this method. Gustafson and Kessel (1979) introduced the first adaptive fuzzy clustering algorithm, based on a quadratic distance defined by a fuzzy covariance matrix. Now these fuzzy *c*-means (FCM) algorithms (Dunn 1974; Bezdek 1980; Gustafson and Kessel 1979) are very popular and has been applied successfully in many areas such as taxonomy, image processing (Kawano et al. 2009), information retrieval, data mining, etc.

FCM algorithm starts with an initial guess for the cluster centers, which intends to mark the mean location of each cluster. The initial guess for these cluster centers is most likely incorrect. Additionally, FCM assigns every data point a membership grade for each cluster. By iteratively updating the cluster centers and the membership grades for each data point, FCM iteratively moves the cluster centers to the "right" location within a data set. This iteration is based on minimizing an objective function that represents the distance from any given data point to a cluster center weighted by that data point's membership grade. Namely

$$\min J\left(U, V\right) = \sum_{i=1}^{c} \sum_{k=1}^{l} u_{ik}^{m} d\left(x_k, v_i\right) \tag{1}$$

$$\text{s.t. } \sum_{i=1}^{c} u_{ik} = 1,\ 0 \le u_{ik} \le 1 \tag{2}$$

where c is the number of clusters and selected as a specified value in this paper, l is the number of data points, $u_{ik} \in [0, 1]$ denotes the degree to which the sample x_k belongs to the ith cluster, m is the fuzzy parameter controlling the speed and achievement of clustering, $d\left(x_k, v_i\right) = \|x_k - v_i\|^2$ denotes the distance between point x_k and the cluster center v_i, and V is the set of cluster centers or prototypes ($v_i \in R^p$).

2.2 The Mahalanobis distance measure

The FCM algorithm selects a well-suited distance measure between the sample and the cluster center. The Euclidean distance is a traditional choice. Given two data points $x_p \in R^n$ and $x_q \in R^n$, their Euclidean distance can be calculated as follows:

$$d_E\left(x_p, x_q\right) = \left\|x_p - x_q\right\|^2 \tag{3}$$

Considering that samples are locally correlated, a local distance measure incorporating samples' correlation might be a better choice as a distance measure. Mahalanobis distance can take into account the covariance among the variables in calculating distances, therefore it appears more suitable to measure the distance (Kawano et al. 2009).

Let X be a $l \times n$ input matrix containing l random observations $x_i \in R^n$, i= 1,...,l. The Mahalanobis distance d_M from a sample x_i to the population X is defined as follows:

$$d_M\left(x_i, X\right) = \left(x_i - \mu\right)^T \sum^{-1} \left(x_i - \mu\right) \tag{4}$$

where μ is a mean vector of all samples, $\sum$ is the covariance matrix calculated as:

$$\sum = \frac{1}{l} \sum_{j=1}^{l} \left(x_j - \mu\right)\left(x_j - \mu\right)^T \tag{5}$$

Originally, the Mahalanobis distance can be defined as a dissimilarity measure between two random vectors of the same distribution with covariance matrix $\sum$. If the covariance matrix is the identity matrix, the Mahalanobis distance reduces to the Euclidean distance.

2.3 Modified FCM algorithm based on Mahalanobis distance

This section presents a modified FCM algorithm based on Mahalanobis distance. To make use of the FCM algorithm for the fuzzy support vector classifier, we extend the original FCM algorithm into the modified algorithm by replacing Euclidean distance with Mahalanobis distance.

Modified FCM algorithm based on Mahalanobis distance minimizes the following objective function

$$\min J\left(U, V, \sum\right) = \sum_{i=1}^{c} \sum_{k=1}^{l} u_{ik}^{m} \left(x_k - \mu_i\right)^T \sum^{-1} \left(x_k - \mu_i\right) \tag{6}$$

$$\text{s.t.} \sum_{i=1}^{c} u_{ik} = 1,\ 0 \le u_{ik} \le 1 \tag{7}$$

The formulation of optimization using Lagrange multiplier method is as following:

$$L = \sum_{i=1}^{c} \sum_{k=1}^{l} u_{ik}^{m} \left(x_k - \mu_i\right)^T \sum\nolimits^{-1} \left(x_k - \mu_i\right) + \sum_{k=1}^{l} \alpha_k \left(1 - \sum_{i=1}^{c} u_{ik}\right) \tag{8}$$

The corresponding dual is found by differentiating with respect to u_{ik}, μ_i and $\sum$ according to (8). We obtain

$$\frac{\partial L}{\partial u_{ik}} = 0 \quad \Rightarrow \; u_{ik} = \left[\sum\nolimits_{j=1}^{c} \left(\frac{\left(x_k - \mu_i\right)^T \sum^{-1} \left(x_k - \mu_i\right)}{\left(x_k - \mu_j\right)^T \sum^{-1} \left(x_k - \mu_j\right)} \right)^{\frac{1}{m-1}} \right]^{-1} \tag{9}$$

$$\frac{\partial L}{\partial \mu_i} = 0 \quad \Rightarrow \; \mu_i = \frac{\sum_{k=1}^{l} u_{ik}^{m} x_k}{\sum_{k=1}^{l} u_{ik}^{m}} \tag{10}$$

$$\frac{\partial L}{\partial \sum} = 0 \quad \Rightarrow \; \sum = \frac{\sum_{k=1}^{l} u_{ik}^{m} \left(x_k - \mu_i\right) \left(x_k - \mu_i\right)^T}{\sum_{k=1}^{l} u_{ik}^{m}} \tag{11}$$

The modified FCM algorithm is implemented by iteratively updating (9) and (10) until the stopping criterion is satisfied.

3 Fuzzy classifier based on modified FCM algorithm

Support vector data description (SVDD) (Tax and Duin 1999, 2004) has been shown effectiveness to detect outliers from the normal data points. However, it is well-known that SVDD method is particular in the case of one-class classification problem, and the training process is very sensitive to outlier and noise data. In order to decrease the effect of those outliers or noises, researchers assign each data point in the training dataset with a membership and sum the deviations weighted by their memberships. If one data point is detected as an outlier, it is assigned with a low membership, so its contribution to total error term will be decreased. Unlike the equal treatment in standard SVM, this method fuzzifies the penalty term to reduce the sensitivity of less important data points.

In this section, we introduce support vector data description, and present fuzzy support vector classifier based on modified FCM algorithm.

3.1 Support vector data description

Over the last several years, SVM has already been generalized to solve one-class problems. Tax and Duin (1999, 2004) presented a support vector data description (SVDD) method for classifier designation and outlier detection. The main idea in Tax and Duin (1999, 2004) is to map data points by means of a nonlinear transformation to a high dimensional feature space and to find the smallest sphere that contains most of the mapped data points in the feature space. This sphere, when mapped back to the data space, can be separated into several components, each enclosing a separate cluster of points. In real operations, they introduce the soft idea and kernel techniques from SVM and allow some data points outside the more general sphere. The advantage of SVDD lies in the fact that the algorithm is very intuitive and comprehensive.

More specifically, given a set of training data $\{x_i,\ i = 1, 2, \cdots, l\} \subset \Re^n$, the minimum enclosing sphere S, which is characterized by its center c and radius R, can be found by solving the following constrained quadratic optimization problem:

$$\min_{c,R} \ R^2 \tag{12}$$

$$\text{s.t. } \|x_i - c\|^2 \leq R^2 \tag{13}$$

However, in real-world applications, training data of a class is rarely distributed spherically, even if the outermost samples are excluded. To have more flexible descriptions of a class, one can first transform the training samples into a high-dimensional feature space using a nonlinear mapping Φ and then compute the minimum enclosing sphere S in the feature space. Furthermore, to allow for the possibility of some samples falling outside of the sphere, one can relax the constraints (13), which require the distance from x_i to the center c to be smaller than R for all x_i, with a set of soft constraints

$$\begin{aligned} &\|\Phi(x_i) - c\|^2 \leq R^2 + \xi_i \\ &\xi_i \geq 0, \ \text{for } i = 1, 2, \cdots, l \end{aligned} \tag{14}$$

where c is the center of the minimum enclosing sphere S and ξ_i are slack variables allowing for soft boundaries. To penalize large distances to the center of the sphere, the quadratic optimization problem in (12), accordingly, is represented as follows.

$$\min_{c,R} \ R^2 + C \sum_i \xi_i \tag{15}$$

under the constraints (14), where $C > 0$ is a penalty constant that controls the trade-off between the size of the sphere and the number of samples that possibly fall outside of the sphere.

To solve this problem, we introduce the Lagrangian

$$L = R^2 - \sum_i \left(R^2 + \xi_i - \|\Phi(x_i) - c\|^2\right) \alpha_i - \sum_i \beta_i \xi_i + C \sum_i \xi_i \tag{16}$$

Using the Lagrange multiplier method, the above constrained quadratic optimization problem can be formulated as the Wolfe dual form.

To solve the optimization problem, we differentiate with respect to R,c and ξ_i according to (16), and set their derivative values are zero, respectively. Namely,

$\frac{\partial L}{\partial R}=0$, $\frac{\partial L}{\partial c}=0$, and $\frac{\partial L}{\partial \xi_i}=0$. We can obtain $\sum_i \alpha_i = 1$, $c=\sum_i \alpha_i \Phi(x_i)$, and $C = \alpha_i + \beta_i$.

Substituting these back into (16), we obtain the dual formulation

$$\begin{aligned} &\max\ W = \sum_i \alpha_i K(x_i, x_i) - \sum_{i,j} \alpha_i \alpha_j K\left(x_i, x_j\right) \\ &\text{s.t. } 0 \le \alpha_i \le C,\ \sum_i \alpha_i = 1,\ \ j = 1, 2, \cdots, l \end{aligned} \tag{17}$$

where the kernel $K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j)$ satisfies Mercer's condition. Points x_i with $\alpha_i > 0$, called support vectors (SVs), are needed in the description and lie on the boundary of the sphere.

To test the point x, the distance to the center of the sphere has to be calculated. A test point x is accepted when this distance is smaller or equal than the radius:

$$\|\Phi(x) - c\|^2 = K(x, x) - 2\sum_i \alpha_i K(x_i, x) + \sum_{i,j} \alpha_i \alpha_j K\left(x_i, x_j\right) \le R^2 \tag{18}$$

where R^2 is the distance from the center of the sphere c to any support vector x_i on the boundary.

3.2 Fuzzy classifier based on support vector data description

Let $\{(x_i, m_i),\ i = 1, 2, \cdots, l\} \subset \Re^n$ be a given training data set, where m_i is the weights of training sample x_i. Once we compute weights m_i of the training sample using the improved FCM method in Section 2, fuzzy support vector classifier problems can be subsequently described as follows:

$$\min_{c,R}\ R^2 + C\sum_i m_i \xi_i \tag{19}$$

$$\text{s.t.}\ \begin{aligned} &\|\Phi(x_i) - c\|^2 \le R^2 + \xi_i \\ &\xi_i \ge 0,\ \text{for } i = 1, 2, \cdots, l \end{aligned} \tag{20}$$

where m_i are the weights generalized by improved FCM method in Section 2.

We can find the solution to this optimization problem in dual variables by finding the saddle point of the Lagrange. Its corresponding Lagrangian formula is

$$L = R^2 - \sum_i \left(R^2 + \xi_i - \|\Phi(x_i) - c\|^2\right)\alpha_i - \sum_i \beta_i \xi_i + C\sum_i m_i \xi_i \tag{21}$$

To solve the optimization problem, we differentiate with respect to R,c and ξ_i according to (21), and set their derivative values are zero, respectively. We obtain $\partial L/\partial R = 2R(1 - \sum_i \alpha_i) = 0$, $\partial L/\partial c = 2\sum_i \alpha_i (\Phi(x_i) - c) = 0$,

$\partial L/\partial \xi_i = Cm_i - \alpha_i - \beta_i = 0$, respectively, which lead to $\sum_i \alpha_i = 1$, $c = \sum_i \alpha_i \Phi(x_i)$, and $Cm_i = \alpha_i + \beta_i$.

Substituting these back into (21), we obtain the dual formulation

$$\begin{aligned} \max\ W = & \sum_i \alpha_i K(x_i, x_i) - \sum_{i,j} \alpha_i \alpha_j K\left(x_i, x_j\right) \\ \text{s.t. } 0 \le \alpha_i \le Cm_i, & \sum_i \alpha_i = 1,\ \ j = 1, 2, \cdots, l \end{aligned} \tag{22}$$

We replace the inner products in the original form (22) by a kernel function $K(x_i, x_j) = \Phi(x_i)\cdot\ \Phi(x_j)$. Several kernel functions have been proposed, including linear kernel, polynomial kernel, radial basis function kernel and sigmoid kernel. In our experiments we use a Gaussian kernel function.

Solving the dual QP problem, we obtain the Lagrange multipliers α_i, which give the center c of the minimum enclosing sphere S as a linear combination of x_i,

$$c = \sum_i \alpha_i x_i \tag{23}$$

According to the Karush–Kuhn–Tucker (KKT) optimality conditions, we have

$$\begin{aligned} \alpha_i = 0 \ &\Rightarrow\ \|\Phi(x_i) - c\|^2 < R^2 \text{ and } \xi_i = 0 \\ 0 < \alpha_i < Cm_i \ &\Rightarrow\ \|\Phi(x_i) - c\|^2 = R^2 \text{ and } \xi_i = 0 \\ \alpha_i = Cm_i \ &\Rightarrow\ \|\Phi(x_i) - c\|^2 \ge R^2 \text{ and } \xi_i \ge 0 \end{aligned} \tag{24}$$

Therefore, only α_i that correspond to training examples x_i that lie either on or outside of the sphere are nonzero. All the remaining α_i are zero and the corresponding training samples are irrelevant to the final solution.

It is clear that the only difference between standard SVDD and fuzzy SVDD is the upper bounds of Lagrange multipliers α_i in the dual problem. In standard SVDD, the upper bounds of α_i are bounded by a constant C while they are bounded by dynamical boundaries that are weight values Cm_i in fuzzy support vector classifier.

3.3 The algorithm description

In this section, we present a method for fuzzy classifier problems. Given a set of training data $(x_1, y_1), (x_2, y_2), \cdots, (x_l, y_l)$, where $l \in R$ denotes the number of training samples, $x_i \in \Re^n$ denotes an input pattern, and $y_i \in R = \{1, 2, \cdots, c\}$ denotes its output class, where c is the number of classes. The central idea of the proposed method is to use the membership matrix generated by the modified FCM algorithm to generate the new training set, and then to use the proposed fuzzy support vector classifier method to classify a data point via the given decision rule.

The detailed procedure of the proposed method is as follows:

Step 1 Use modified FCM algorithm to compute the membership u_{ik} of each training sample x_i to each class k. Given a threshold σ_t for membership matrix $U = \{u_{ik}\}$, set $u_{ik} = 0$ if $u_{ik} \le \sigma_t$.

Step 1.1 Select the number of clusters c, the maximal iterative count N_{max}, fuzziness parameter m (let $m = 2$), and converge error $\varepsilon > 0$.

Step 1.2 Initialize the membership matrix $U^{(0)} = \left\{ u_{ik}^{(0)} \right\}$ satisfied with constraint conditions $\sum_{i=1}^{c} u_{ik} = 1, 0 \le u_{ik} \le 1$.

Step 1.3 For $t = 1,\ldots,N_{max}$,
calculate the membership matrix $U^{(t)} = \left\{ u_{ik}^{(t)} \right\}$ according to (9);
calculate the cluster centers $\mu_i^{(t)}$ $(i = 1, ..., c)$ according to (10);
calculate the objective function $J^{(t)}(U,V,\sum)$ according to (6);
when $\left| J^{(t)}\left(U, V, \sum\right) - J^{(t-1)}\left(U, V, \sum\right) \right| < \varepsilon$ or $t = N_{max}$, stop iteration and return the membership matrix $U^{(t)}$ and the cluster centers $\mu_i^{(t)}$.

Step 1.4 Modify the membership matrix $U = \{u_{ik}\}$, set $u_{ik} = 0$ if $u_{ik} \le \sigma_t$.

After partitioning, the weights of outliers and noises will be very small or even nearly to zero, such that the effect of outliers and noises to decision hyperplane is significantly reduced. To reduce the training time in the fuzzy support vector classifier, we set a threshold σ_t for weights u_{ik}. If $u_{ik} > \sigma_t$, it denotes the training sample x_k affects the ith class to some degree. Otherwise, training algorithm ignores the effect of x_k to the ith class.

Step 2 Generate the new training set, and partition the given training data into c subsets $D_p(p = 1,\ldots,c)$ according to their membership u_{ik}.

Step 3 For each class data set D_p, train their data points using the proposed fuzzy support vector classifier method. Let the solution of the dual problem (9) be $\alpha_i^*, i = i_1, \cdots, i_{l_p}$, and $J_p \subset \left\{1, \cdots, l_p\right\}$ be the set of the index of the nonzero α_i^*. The trained kernel support function for each class data set D_p is then given by

$$f_p(x) = K(x, x) - 2 \sum_{i \in J_p} \alpha_i^* K(x_i, x) + \sum_{i, j \in J_p} \alpha_i^* \alpha_j^* K\left(x_i, x_j\right) \tag{25}$$

Step 4 Construct the following classification rule for the new sample x:

$$F(x) = \arg \max_p \frac{l_p}{l} \left(\left(R_p^* \right)^2 - f_p(x) \right) \tag{26}$$

where $p = \{1, 2, \cdots, C\}$ $p = \{1, 2, \cdots, C\}$, $\left(R_p^*\right)^2 = f_p(x_i)$ for any support vector x_i of the pth class, $l \in R$ denotes the number of the training samples, and $l_p \in R$ denotes the number of the pth class in the training samples.

In step 4, we propose the classification rule based on the SVDD which is an optimal classifier indicated by Lee and Lee (2007) according to the Bayesian decision theory, and show that it leads to a better generalization accuracy than the classification rule in Zhu et al. (2003) and Wang et al. (2007) through experiments in Section 4.

4 Experimental results

The effectiveness of the proposed fuzzy support vector classifier algorithm is tested on benchmark data sets from the UCI machine learning repository (http:/www.ics.uci.edu/~mlearn/MLRepository.html), including *glass, ionosphere, sonar, pima-diabetes, iris* and*vehicle*. Their related parameters are listed in Table 1. In the Table 1, #*pts* denotes the number of data points, #*att* denotes the number of the attributes, and #*class* denotes the number of the classifications.

On each data set, we compared our method to the standard SVM classifier and to the sphere-based classifier using the decision rule of Zhu et al. (2003) and Wang et al. (2007). We used Gaussian kernel for all algorithms and used the ten-fold cross-validation method to estimate the generalization accuracy of the classifiers. In the cross validation process, we ensured that each training set and each testing set were the same for all algorithms.

Accordingly, let a threshold σ_t be 0.3 in the step 1 of our proposed algorithm. The value of the threshold σ_t influences the algorithm runtime. For small values of σ_t more training samples may be included in the extra class.

In the experiments we standardized the data to zero mean. While using the Gaussian kernel, we determined the tuning parameters σ^2 and C with standard ten-fold cross validation. The initial search for optimal parameters σ^2 and C was done on a 10×10 uniform coarse grid in the (σ^2, C) space, namely, $\sigma^2 = [2^{-3}, 2^{-1}, \ldots, 2^{13}, 2^{15}]$, $C = [2^{-15}, 2^{-13}, \ldots, 2^1, 2^3]$.

The experimental results are summarized in Table 2. As we see from Table 2, except on the *sonar* and *iris* data sets, our proposed algorithm improves significantly in classification accuracy and is better than that of the standard SVM classifier on four out of the six data sets being tested. Moreover, our proposed method also improves when compared with the sphere-based classifier in Zhu et al. (2003) and Wang et al. (2007) from experimental results.

Additionally, to test algorithm sensitivity to outlier data, we random add 10% "outlier" data points to iris and vehicle, respectively, which are labeled as *iris** and v*ehicle** in Table 2. Experimental results are shown as the last 2 columns in Table 2. Results indicate that the proposed method successfully reduce the effect of outliers and accordingly yield good generalization performance, especially existing outlier data in data sets.

On the other hand, the proposed method is smaller than SVM method in time costs, whereas it is trivially higher than sphere-based classifier method.

Parameter σ_t has some effect on the prediction accuracy. Through our experiments, we find when the value of σ_t is bigger (for example $\sigma_t > 0.4$), the accuracy of our proposed method is inferior to other methods, whereas its time cost is smaller because the training data points decreases. So, we select σ_t be 0.3 in our experiments.

Table 1 Data sets from UCI

Data set	#*pts*	#*att*	#*class*
Glass	214	9	6
Ionosphere	351	34	2
Sonar	208	60	2
Pima-diabetes	768	8	2
Iris	150	4	3
Vehicle	846	18	4

Table 2 Experimental results

Methods		Data set							
		Glass	Ionosphere	Sonar	Pima-diabetes	Iris	Vehicle	Iris*	Vehicle*
SVM	Accuracy (%)	72.4	94.9	89.4	72.5	97.3	87.4	92.6	76.5
	Training time (s)	7.6	12.5	8.5	32.4	0.2	48.4	0.3	50.8
Sphere-based classifier	Accuracy (%)	71.6	95.2	84.9	73.0	97.3	87.4	91.7	76.6
	Training time (s)	3.5	8.6	5.4	19.6	0.2	32.7	0.2	34.7
Proposed method ($\sigma_t = 0.3$)	Accuracy (%)	72.6	95.4	88.6	73.1	97.3	87.5	93.4	78.9
	Training time (s)	5.5	11.0	8.0	23.1	1.3	38.2	1.4	39.4

5 Conclusions

In this paper, a fuzzy support vector classifier has been developed using fuzzy *c*-means clustering based on Mahalanobis distance. The proposed algorithm can be used to deal with the outlier sensitivity problem in traditional multi-class classification problems. The modified fuzzy *c*-means clustering algorithm based on Mahalanobis distance takes into the samples' correlation account, and is improved to generate different weight values for main training data points and outliers according to their relative importance in the training data. Experimental results show that the proposed method can reduce the effect of outliers and give higher classification accuracy rate than other existing methods do.

Acknowledgements This work is supported by the Liaoning Doctoral Research Foundation of China (Grant No. 20081079), and the University Scientific Research Project of Liaoning Education Department of China (Grant No. 2008347). The authors are grateful for the anonymous reviewers who made constructive comments.

References

Bezdek, J. C. (1980). A convergence theorem for the fuzzy ISODATA clustering algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 2*, 1–8.

Dunn, J. C. (1974). A fuzzy relative to the ISODATA process and its use in detecting compact, well-separated clusters. *Journal of Cybernetics, 3*, 32–57.

Gustafson, D. E., & Kessel, W. C. (1979). Fuzzy clustering with a fuzzy covariance matrix. In: *Proceedings of the IEEE Conference on Decision and Control* (pp. 761–766). San Diego, CA.

Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multi-class support vector machines. IEEE *Transactions on Neural Networks, 13*(2), 415–425.

Hu, W. J., & Song, Q. (2004). An accelerated decomposition algorithm for robust support vector machines. *IEEE Transactions on Circuits and Systems II: Express Briefs, 51*(5), 234–240.

Inoue, T., & Abe, S. (2001). Fuzzy support vector machines for pattern classification. In: *Proceedings of International Joint Conference on Neural Networks (IJCNN '01)* (Vol. 2, pp. 1449–1454).

Kawano, H., Suetake, N., Cha, B., & Aso, T. (2009). Sharpness preserving image enlargement by using self-decomposed codebook and Mahalanobis distance. *Image and Vision Computing, 27*(6), 684–693.

Lee, D., & Lee, J. (2007). Domain described support vector classifier for multi-classification problems. *Pattern Recognition, 40*, 41–51.

Lin, C. F., & Wang, S. D. (2002). Fuzzy support vector machines. *IEEE Transactions on Neural Networks, 13*(2), 464–471.

Lin, C. F., & Wang, S. D. (2003). Training algorithms for fuzzy support vector machines with noisy data. In *Proceedings of the IEEE 8th Workshop on Neural Networks for Signal Processing* (pp. 517–526).

Platt, J. C., Cristianini, N., & Shawe-Taylor, J. (2002). Large margin DAG's for multiclass classification. In S. A. Solla, T. K. Leen, & K. R. Müller (Eds.), Advances in neural information processing systems (Vol. 12, pp. 547–553). Cambridge: MIT Press.

Song, Q., Hu, W. J., & Xie, W. F. (2002). Robust support vector machine with bullet hole image classification. *IEEE Trans. on Systems, Man and Cybernetics, 32*(4), 440–448.

Tax, D., & Duin, R. (1999). Support vector domain description. *Pattern Recognition Letters, 20*, 1191–1199.

Tax, D., & Duin, R. (2004). Support vector data description. *Machine Learning, 54*, 45–66.

Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York: Springer.

Wang, J., Neskovic, P., & Cooper, L. N. (2007). Bayes classification based on minimum bounding spheres. *Neurocomputing, 70*, 801–808.

Weston, J., & Watkins, C. (1998). Multi-class support vector machines, department of computer science, Royal Holloway University of London Technical Report, SD2TR298204.

Yu, H., Han, J., & Chang, K. C. (2003). Classifying large datasets using SVMs with hierarchical clusters. In: *Proceedings of the International Conference on Knowledge Discovery and Data Mining (KDD'03)* (pp. 306–315).

Zhang, Y., Chi, Z. X., Liu, X. D., & Wang, X. H. (2007). A novel fuzzy compensation multi-class support vector machines. *Applied Intelligence, 27*(1), 21–28.

Zhu, M. L., Chen, S. F., & Liu, X. D. (2003). Sphere-structured support vector machines for multi-class pattern recognition. *Lecture Notes in Computer Science, 2639*, 589–593.