New Decision Function for Support Vector Data Description

Mohamed EL Boujnouni, Mohamed Jedra, Noureddine Zahid
Faculty of Sciences, Mohammed V – Agdal University
Laboratory of Conception and Systems (Microelectronic and Informatics) Avenue Ibn Battouta B.P 1014
Rabat, Morocco
med_elbouj@yahoo.fr, {jedra, zahid}@fsr.ac.ma

ABSTRACT: *In conventional support vector data description (SVDD), for each class we look for the smallest sphere that encloses its data. in the decision phase a sample is classified into class i only when the value of the i^{th} decision function is positive. Following this architecture, an unclassifiable region (s) can be appeared if the values of more than one decision function are positives. To overcome this problem, we propose a new simple and powerful decision function, which is used only in the overlapped regions, this membership function can be calculated in the feature space and can be represented by kernels functions. This method gives good performance on reducing the effects of overlap and significantly improves the classification. We demonstrate the performance of our decision function using six benchmark datasets.*

Keywords: SVDD, Decision Function, Membership, Overlaps

Received: 12 June 2012, Revised 28 July 2012, Accepted 5 August 2012

1. Introduction

In domain description the task is not to distinguish between classes of objects like in clustering problems, or to produce a desired outcome for each input object like in regression problems, but to give a description of a set of objects. This description should be able to distinguish between the classes of objects represented by the training set, and all other possible objects in the object space [1, 8]. Recently, a Support Vector Data Description (SVDD) (also called One-class Classification). Inspired by support vector machine was invented by Tax and Duin [1]. In a SVDD the compact description of target data is given as a hyper sphere with minimal volume containing most of normal data, rejecting most of negative data, in a high-dimensional feature space using some kernel functions Table 1, [4]. SVDD uses support vectors to describe the boundary of target class as Support Vector Machine SVM does [5]. Support vectors are found by solving convex quadratic programming (QP) problem. But after finding the hyper spheres which characterize the classes, a new problem can be appeared. This problem is linked to the probable overlap of these hyper spheres, and influences directly the decision function which gives more than one class for the same test object. To resolve this problem we propose to substitute the standard decision function by a new one only in the overlapped regions, this idea is inspired by the notion of the fuzzy membership which has been investigated for a period of time [10-16] for SVM to make the classification become more effective. Lin and Wang (2002) [13] first proposed a prototype of fuzzy SVM , where one applies a fuzzy membership function to each input data of SVM, this function denotes the attitude of the corresponding point toward one class. In other hand on the fuzzy multi-classification, many researchers have discussed the unclassifiable regions [17-21] existing in SVM and introduced a fuzzy membership functions.

In this paper we introduce a new decision function, which quantify the degree of membership for a sample localized in the intersection volume. Computer simulations have been conducted on six benchmark datasets to demonstrate the effectiveness of the proposed method. In the following section we give the theoretical background of the two version of SVDD, in section 3 we focus on the standard decision function, in section 4 we introduce a fuzzy membership used in SVM, in section 5 we present the new decision function for SVDD. In the last section we give several experiments results to show the validity of our proposed algorithm.

2. Support Vector Data Description (SVDD)

The normal data description model [2, 3] gives a closed boundary around the data: a hypersphere characterized by center a and radius R > 0. It minimizes the volume of the sphere by minimizing R^2, and demand that the sphere contains all training objects x_i. Let $x_i \in \chi$ be a data set of N points, with, $x_i \in R_d$ the data space, we look for the smallest enclosing sphere of radius R which is described by the following constraints:

$$\| x_j - a \|^2 \leq R^2 \ \forall j \tag{1}$$

Where || . || is the Euclidean norm. Soft constraints are incorporated by adding slack real and positive variable ε_j:

$$\| x_j - a \|^2 \leq R^2 + \varepsilon_j \ \forall j \tag{2}$$

To solve this problem we introduce the Lagrangian:

$$L = R^2 - \sum_j (R^2 + \varepsilon_j - \| x_j - a \|^2) \alpha_j - \sum_j \varepsilon_j \mu_j + C \sum_j \varepsilon_j \tag{3}$$

Where $\alpha_j \geq 0$ and $\mu_j \geq 0$ are Lagrange multipliers, C is a constant, and $C \Sigma_j \varepsilon_j$ is a penalty term. Setting the partial derivatives of L with respect to R, a, ε_i to zero gives the following constraints:

$$\frac{\partial L}{\partial R} = 0 \Rightarrow \sum_j \alpha_i = 1 \tag{4}$$

$$\frac{\partial L}{\partial a} = 0 \Rightarrow a = \sum_j \alpha_i x_i \tag{5}$$

$$\frac{\partial L}{\partial \varepsilon_i} = 0 \Rightarrow \alpha_i = C - \mu_i \tag{6}$$

The solution of the primal problem can be obtained by solving its dual problem [2]
Max:

$$W = \sum_j x_j^2 \alpha_j - \sum_{i,j} \alpha_i \alpha_j x_i x_j \tag{7}$$

$$Subject\ to\ 0 \leq \alpha_j \leq C \ \forall j\ and \sum_j a_j = 1$$

When negative examples (objects which should be rejected) are available, they can be incorporated in the training to improve the description. In contrast with the training (target) examples which should be within the sphere, the negative examples should be outside it. In the following, the target objects are enumerated by indices *i, j* and the negative examples by *l, m*. Again we allow for errors in both the target and the outliers set and introduce slack real positive variables ε_i and ε_l [2]:

$$L(R, a, \varepsilon_i, \varepsilon_l) = R^2 + C1 \sum_i \varepsilon_l + C2 \sum_l \varepsilon_l \tag{8}$$

With the constraints:

$$\| x_i - a \|^2 \leq R^2 + \varepsilon_i \quad \| x_l - a \|^2 \geq R^2 - \varepsilon_l \qquad \varepsilon_i, \varepsilon_l \geq 0 \ \forall i, l$$

Where $C1$, $C2$ are constants real positives, $C1 \Sigma_i \varepsilon_i$ and, $C2 \Sigma_i \varepsilon_i$ are penalty terms, these constraints are incorporated in equation (8) and the Lagrange multipliers $\alpha_i, \alpha_l, \gamma_i, \gamma_l$ are introduced as follow:

$$L(R, a, \varepsilon_i, \varepsilon_l, \alpha_i, \alpha_l, \gamma_i, \gamma)$$

$$= R^2 + C1\sum_i \varepsilon_i + C2\sum_l \varepsilon_l - \sum_i \gamma_i \varepsilon_i - \sum_l \gamma_l \varepsilon_l - \sum_i \alpha_i [R^2 + \varepsilon_i - \| x_i - a \|^2] \tag{9}$$

$$-\sum_l \alpha_l [\| x_l - a \|^2 - R^2 + \varepsilon_l]$$

With $\alpha_i \geq 0$, $\alpha_l \geq 0$, $\gamma_l \geq 0$, $\gamma_l \geq 0$ are Lagrange multipliers. Setting the partial derivatives of L with respect to R, a, ε_i and ε_l to zero gives the following constraints:

$$\frac{\partial L}{\partial R} = 0 \Rightarrow \sum_i \alpha_i - \sum_j \alpha_l = 1 \tag{10}$$

$$\frac{\partial L}{\partial a} = 0 \Rightarrow \mathrm{a} = \sum_i \alpha_i x_i - \sum_l \alpha_l x_l \tag{11}$$

$$\frac{\partial L}{\partial \varepsilon_i} = 0 \; and \; \frac{\partial L}{\partial \varepsilon_l} = 0 \; and \; \Rightarrow \alpha_i = C1 - \gamma_i \quad \alpha_l = C2 - \gamma_l \; \forall i, l \tag{12}$$

When Equations (10), (11) are substituted into Equation (9) we obtain :

Max

$$W = \sum_i \alpha_i x_i x_i - \sum_l \alpha_l x_l x_l - \sum_{i,j} \alpha_i \alpha_j x_i x_j + 2\sum_{i,j} \alpha_l \alpha_j x_l x_j - \sum_{l,m} \alpha_l \alpha_m x_l x_m \tag{13}$$

Subject to:

$$0 \leq \alpha_i \leq C1 \; and \; 0 \leq \alpha_l = C2 \; \forall i, l \; and \; \sum_i \alpha_i - \sum_l \alpha_l = 1$$

The formulations of SVDD can be extended to obtain a more flexible description. Data is mapped nonlinearly into a higher dimensional space where a hyperspherical description can be found. The mapping is performed implicitly, replacing all of the inner products by a kernel function K (x_i, x_j) [2, 3]. Table 1 describes some commonly used kernel functions:

Gaussian Radial Basis Function	$k(x,y) = e^{\left(-(x-y)^2 / 2\sigma^2\right)}$
Exponential Radial Basis Function	$k(x,y) = e^{\left(-\lvert x-y \rvert / 2\sigma^2\right)}$
Hyperbolic Tangent	$k(x,y) = \tanh(b(x,y) + c)$
Fourier Series	$k(x,y) = \dfrac{\sin(\delta + \frac{1}{2}(x-y))}{\sin(\frac{1}{2}(x-y))}$
Linear Spliness	$k(x,y) = 1 + x \,.\, y + x \,.\, y.min(x,y) - \dfrac{(x+y)}{2} min(x,y))^2 + \dfrac{1}{3} max(x,y))^3$
Bn-splines	$k(x,y) = \mathrm{B}_{2n+1}(\mathrm{x} - \mathrm{y})$
Polynomial	$k(x,y) = (1 + x.y)^p$
Two-layer perception	$k(x,y) = \tanh(s_0 \,.\, x, y + s_1)$

Table 1. Some commonly used kernel functions

3. Standard decision function

For multiclass problems, to classify a test point z, we just investigate whether it is inside the hypersphere (a_k, R_k) constructed during the training and associated to the class k [2,3,7].Namely the decision function is calculated as equation (14), if its value is positive for the k^{th} class and negative for the others we conclude that z belong to the class k.

$$f(z) = sgn\,(R_k^2 - \|z - a_k\|^2) \tag{14}$$

This function can be calculated as follow:

In the normal data description case we obtain:

$$\|z - a_k\|^2 = z\,.\,z - 2\sum_i \alpha_{ki}\, x_i z + \sum_{i,j} \alpha_{ki}\, \alpha_{kj} x_i\, x_j \tag{15}$$

$$R_k^2 = x\,.\,x - 2\sum_i \alpha_{ki}\, x_i x + \sum_{i,j} \alpha_{ki}\, \alpha_{kj} x_i\, x_j \tag{16}$$

With α_{kj} is the j^{th} Lagrangian multiplier corresponding to the k^{th} class. And $x \in SV$ the set of Support Vectors having $0 < \alpha_i < C$

In the SVDD with negative examples case we obtain:

$$\|z - a_k\|^2 = z\,.\,z - 2\Big(\sum_i \alpha_{ki}\, x_i z - \sum_l \alpha_{kl}\, x_l z\Big) + \sum_{i,j} \alpha_{ki}\, \alpha_{kj} x_i\, x_j + \sum_{l,m} \alpha_{kl}\, \alpha_{km} x_l\, x_m - 2\sum_{i,l} \alpha_{ki}\, \alpha_{kl} x_i\, x_l \tag{17}$$

$$R_k^2 = x\,.\,x - 2\Big(\sum_i \alpha_{ki}\, x_i x - \sum_l \alpha_{kl}\, x_l x\Big) + \sum_{i,j} \alpha_{ki}\, \alpha_{kj} x_i\, x_j + \sum_{l,m} \alpha_{kl}\, \alpha_{km} x_l\, x_m - 2\sum_{i,l} \alpha_{ki}\, \alpha_{kl} x_i\, x_l \tag{18}$$

For any $x \in SV$ the set of support vectors having $0 < \alpha_i < C1$ (with x is a target object) or $0 < \alpha_l < C2$ (with x is negative object).

4. Fuzzy Support Vector Machine (FSVM)

On the basis of the theory of classical SVM, [13, 22] proposed the theory of fuzzy support vector machine. In classical SVM, each sample is treated equally; i.e., each input point is fully assigned to one of the two classes. However, in many applications, some input points, such as the outliers, may not be exactly assigned to one of these two classes, and each point does not have the same meaning to the decision surface. To solve this problem, fuzzy membership to each input point of SVM can be introduced, such that different input points can make different contribution to the construction of decision surface [13].

Suppose the training samples are given as [9] :

$S = \{(X_i, y_i, s_i)\}$, $i = 1,..., N$ Where each $X_i \in R^d$ and $y_i \in$ {-1, +1} represents its class label, s_i (i =1... N) is a fuzzy membership which satisfies $\sigma \le s_i \le 1$ with a sufficiently small constant $\sigma > 0$, it contains two classes.

One class contains such sample point X_i with $y_i = 1$, denoting this class by C^+, then,

$C^+ = \{X_i \,/\, X_i \in S \;\; and \;\; y_i = +1\}$

One class contains such sample point X_i with $y_i = -1$, denoting this class by C^-, then,

$C^- = \{X_i \,/\, X_i \in S \;\; and \;\; y_i = -1\}$

In linear case, [13] proposed a fuzzy membership which can be described as follows:

Denote the mean of class C^+ and class C^- as X_+ and X_-, respectively

The radius of class C^+ is:

$$r^+ = max\, \|X_+ - X_i\| \;\; where\; X_i \in C^+ \tag{19}$$

And the radius of class C^- is:

$$r^{-} = max \| X_{-} - X_i \| \;\; where\; X_i \in C^{-} \tag{20}$$

The fuzzy membership s_i is [13] :

$$s_i = \begin{cases} 1 - \dfrac{\| X_{+} - X_i \|}{r^{+} + \delta} & if\; X_i \in C^{+} \\ 1 - \dfrac{\| X_{-} - X_i \|}{r^{-} + \delta} & if\; X_i \in C^{-} \end{cases} \tag{21}$$

Where $\delta > 0$ is a constant to avoid the case $s_i = 0$.

In nonlinear case, [9] proposed the kernel version of the fuzzy membership described above, as follows:

Let $\Phi\,(X_i)$ be a mapping function from the input space into the feature space, Define Φ_{+} and Φ_{-} as the class center of C^{+} and C^{-}, respectively. By [9]:

$$\Phi_{+} = \frac{1}{n_{+}} \sum_{X_i \in C^{+}} \Phi\,(X_i) \tag{22}$$

$$\Phi_{-} = \frac{1}{n_{+}} \sum_{X_i \in C^{-}} \Phi\,(X_i) \tag{23}$$

Where n_{+} and n_{-} are the number of the samples of class C^{+} and C^{-}, respectively.

Define the radius of C^{+} by

$$r_{+} = max \| \Phi_{+} - \Phi\,(X_i) \| \;\; where\;\; X_i \in C^{+}$$

$$r_{+}^2 = max \left\{ k\,(X', X') + \frac{2}{n_{+}} \sum_{X_i \in C^{+}} k\,(X_i, X') + \frac{1}{n^2} \sum_{X_i \in C^{+}} \sum_{X_j \in C^{+}} k\,(X', X') \right\} where\;\; X_i \in C^{+} \tag{24}$$

And the radius of C^{-} by

$$r_{-} = max \| \Phi_{-} - \Phi\,(X_i) \| \;\; where\;\; X_i \in C^{-}$$

$$r_{-}^2 = max \left\{ k\,(X', X') + \frac{2}{n_{-}} \sum_{X_i \in C^{-}} k\,(X_i, X') \frac{1}{n_{-}^2} \sum_{X_i \in C^{-}} \sum_{X_j \in C^{-}} k\,(X_i, X_j) \right\} where\;\; X' \in C^{-} \tag{25}$$

The square of the distance between a sample $X_i \in C^{+}$ and its class center in the feature space can be calculated as:

$$d_{i+}^2 = \| \Phi\,(X_i) - \Phi_{+} \|^2$$

$$d_{i+}^2 = k\,(X_i, X_i) - \frac{2}{n_{+}} \sum_{X_j \in C^{+}} k\,(X_i, X_j) + \frac{1}{n_{+}^2} \sum_{X_j \in C^{+}} \sum_{X_k \in C^{+}} k\,(X_j, X_k) \tag{26}$$

Similarly, the square of the distance between each $X_i \in C^{-}$ and its class center in the feature space can be calculated as:

$$d_{i+}^2 = \| \Phi\,(X_i) - \Phi_{+} \|^2$$

$$d_{i-}^2 = k\,(X_i, X_i) - \frac{2}{n_{-}} \sum_{X_j \in C^{-}} k\,(X_i, X_j) + \frac{1}{n_{-}^2} \sum_{X_j \in C^{-}} \sum_{X_k \in C^{-}} k\,(X_j, X_k) \tag{27}$$

For each i (i =1,.., N), the proposed fuzzy membership s_i can be described as follows[9]:

$$s_i = \begin{cases} 1 - \sqrt{\| d_{i+}^2 \| / (r_+^2 + \delta) \|} & if\ y_i = +1 \\ 1 - \sqrt{\| d_{i+}^2 \| / (r_+^2 + \delta) \|} & if\ y_i = -1 \end{cases} \tag{28}$$

Where $\delta > 0$ is used to avoid the case $s_i = 0$. Clearly, it is a function of the center and radius of each class in the feature space and represented with kernel.

5. SVDD with the new decision function

For multiclass problems, a test object z belong to a class k, if only the k[th] standard decision function is positive, following this architecture a sample is unclassifiable if the values of more than one decision function are positives or all the values are negatives, we are interested in the first case, which can be explained by the overlap of the hyperspheres called candidates, To deal with this problem, we propose to quantify the membership of z to each of these candidates. By using a new decision function exclusively for samples existing in the overlap volumes.

We propose to use the fuzzy membership described above, which achieves good performance in FSVM following [9, 13, 22], as a new decision function for SVDD, the center and the radius, whose depend on this function, are determined by solving the QP problem (see equation (5, 11, 16, 18)), contrary to the proposed FSVM where we use respectively, the mean of the class and the maximum distance between each input point and its class center,

In linear case the proposed decision function can be written as:

$$s_k = 1 - \frac{\| z - a_k \|}{r_k + \delta} \tag{29}$$

In nonlinear case the proposed decision function can be re-written as:

$$s_k = 1 - \sqrt{\frac{\| z - a_k \|^2}{r_k^2 + \delta}} \tag{30}$$

Where $\delta > 0$ is a constant to avoid the case $s_k = 0$,

In the normal data description case $\| z - a_k \|$ and r_k are defined respectively in equation (15) and (16), in the SVDD with negative examples case and and are defined respectively in eqequation (17) and (18). In nonlinear case we just replace the inner product equation $\| z - a_k \|$ and r_k (15,16,17,18) by a kernel function see Table 1. After calculating the values of these functions for all hyperspheres candidates, in which z exist simultaneously, z is classified into the class:

$$argmax_i (s_i) \tag{31}$$

To show the effect of this novel decision method we ran the SVDD with negative examples on two classes each one contains a different number of data, these points are distributed differently. We remark in Figure 1 that when we use the normal decision function we can not identify the samples existing in the intersection region, because they belong to two classes simultaneously. We observe also that the new function divide this region into two parts each one is integrated into its original cluster which seems a good membership.

6. Experiments and results

6.1 Datasets and Experimental Setting

Six datasets were used to test the new decision function. The datasets describe various characteristics from studying monk-1, monk-2, monk-3, iris flowers, wine, and glass; all of these datasets are taken from the UCI Repository [6]. Further details of these datasets are provided in Table 2.

Firstly, the three problems defined for monk's dataset were used in the experiment; monks-1 is in standard disjunctive normal form and is supposed to be easily learnable by most of the algorithms and decision trees. Conversely, monk's-2 is similar to parity problems. It combines different attributes in a way that makes it complicated to describe using the given attributes only; monks-3 serves to evaluate the algorithms under the presence of noise.

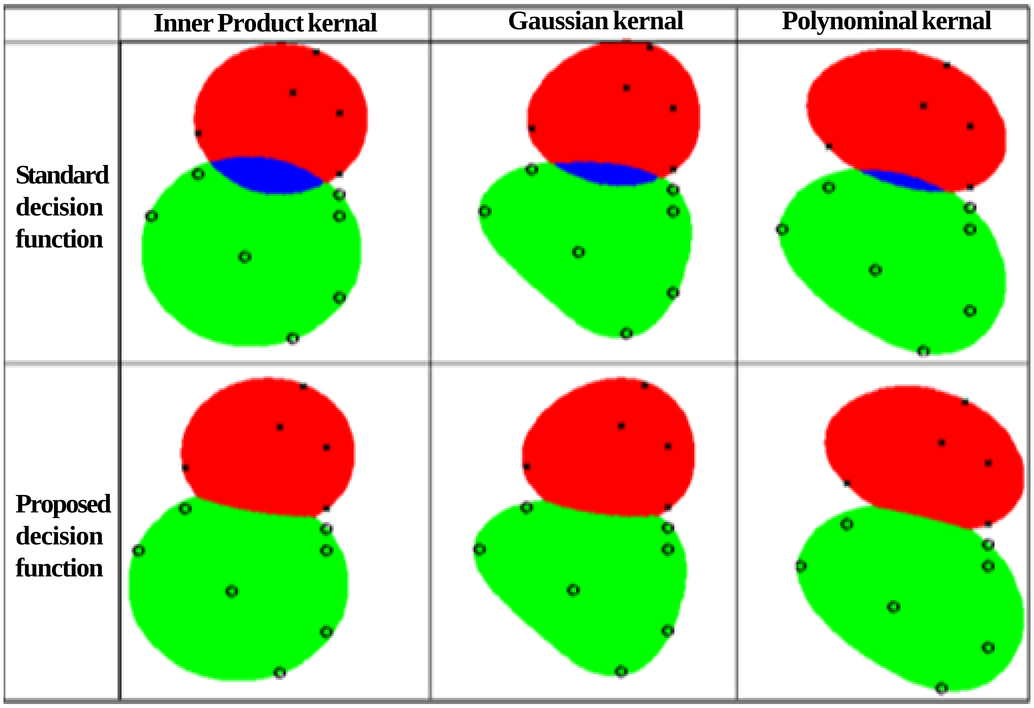


Figure 1. An artificial data set separated with SVDD (C1 = C2 =100) using an inner products, Gaussian kernel (σ = 2) and polynomial kernel (p = 2), for both standard and new decision functions

Secondly, the iris dataset consists of three classes, each of which has 50 samples. While one cluster is easily separable, it is difficult to achieve separation between the other two clusters. Data points correspond to the plants and attributes correspond to sepal and petal measurements.

Thirdly, the wine dataset is the results of a chemical analysis of wines grown in the same region but derived from three different cultivars. The analysis determined the quantities of constituents found in each of the three types of wines.
Fourthly, the glass dataset is the study of classification of types of glass was motivated by criminological investigation data points correspond to the type of glass and attributes correspond to their oxide content (i.e. Na, Fe, K, etc).

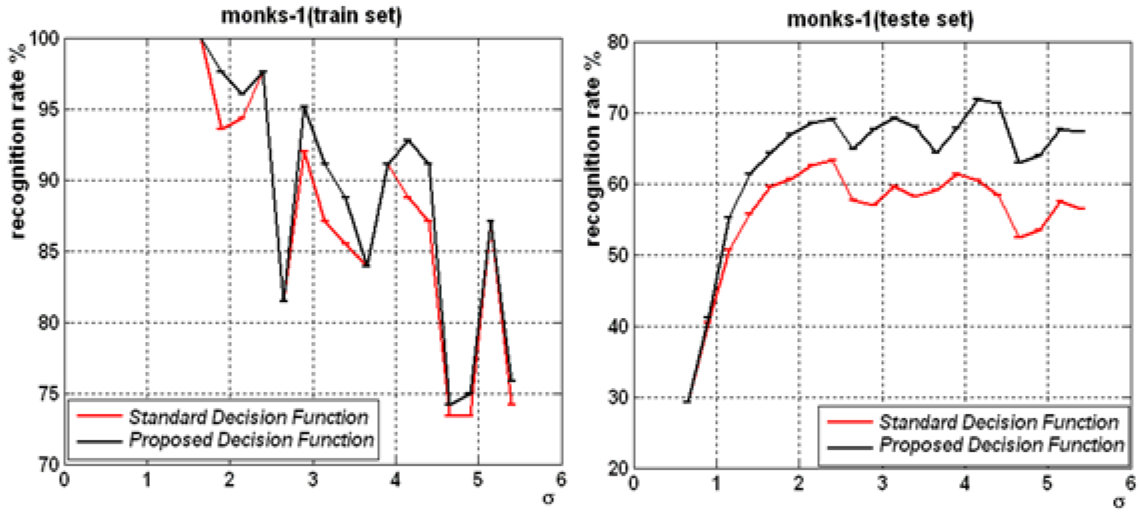
For monk's problem we use the files monks-(1, 2, 3).train, as training set and their corresponding files monks-(1, 2, 3).test, as testing set. For iris, wine, and glass datasets, we randomly split each one into 20 subsets, each subset contains training and testing sets, with the scheme described in Table 2.Training and test sets do not intersect.

6.2 Numerical Results

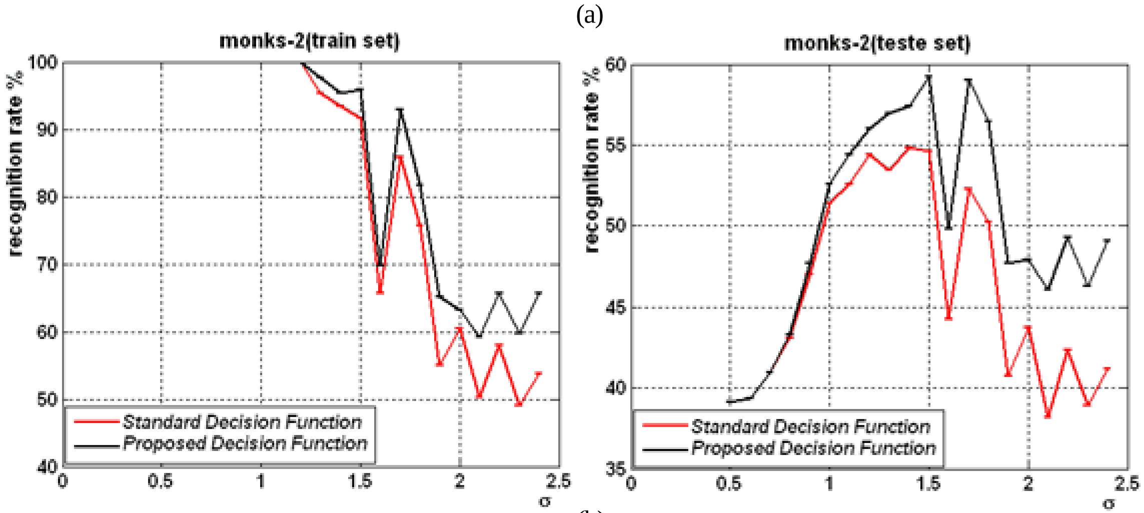
For iris, wine, and glass: To test a dataset, we select the values of ó (RBF width), we fix C =1000, then for each subset of this dataset, SVDD will be trained by the training dataset and then, tested by the training and the testing dataset, using respectively the new decision function equation (31), and the standard one. After terminating the 20 experiments, we calculate the average of the recognition rate and the standard deviation for each decision function.

For monks (1 2 3) datasets: We select the values of σ, we fix C =1000, SVDD will be trained by monks-*x*.train and then, tested by monks-*x*.train, and monks-*x*.test, using respectively the new decision function equation (31), and the standard one, *x* takes the values {1, 2, 3}.we calculate the recognition rate for each decision function.

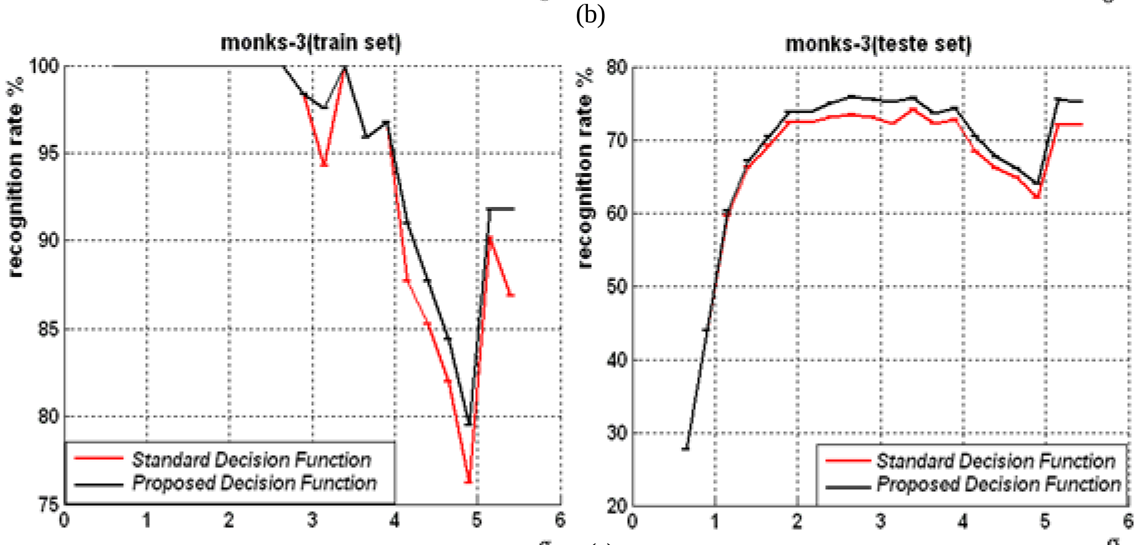
We remark that (see Figure 1): In most of cases, the recognition rate using the new decision function is higher than the rate with the standard one, in the remaining cases, they achieve the same performance. We observe also that generally, the values of the standard deviation are lowers when using the new decision function. This result proves the performance and the stability of our new decision function.



(a)



(b)



(c)

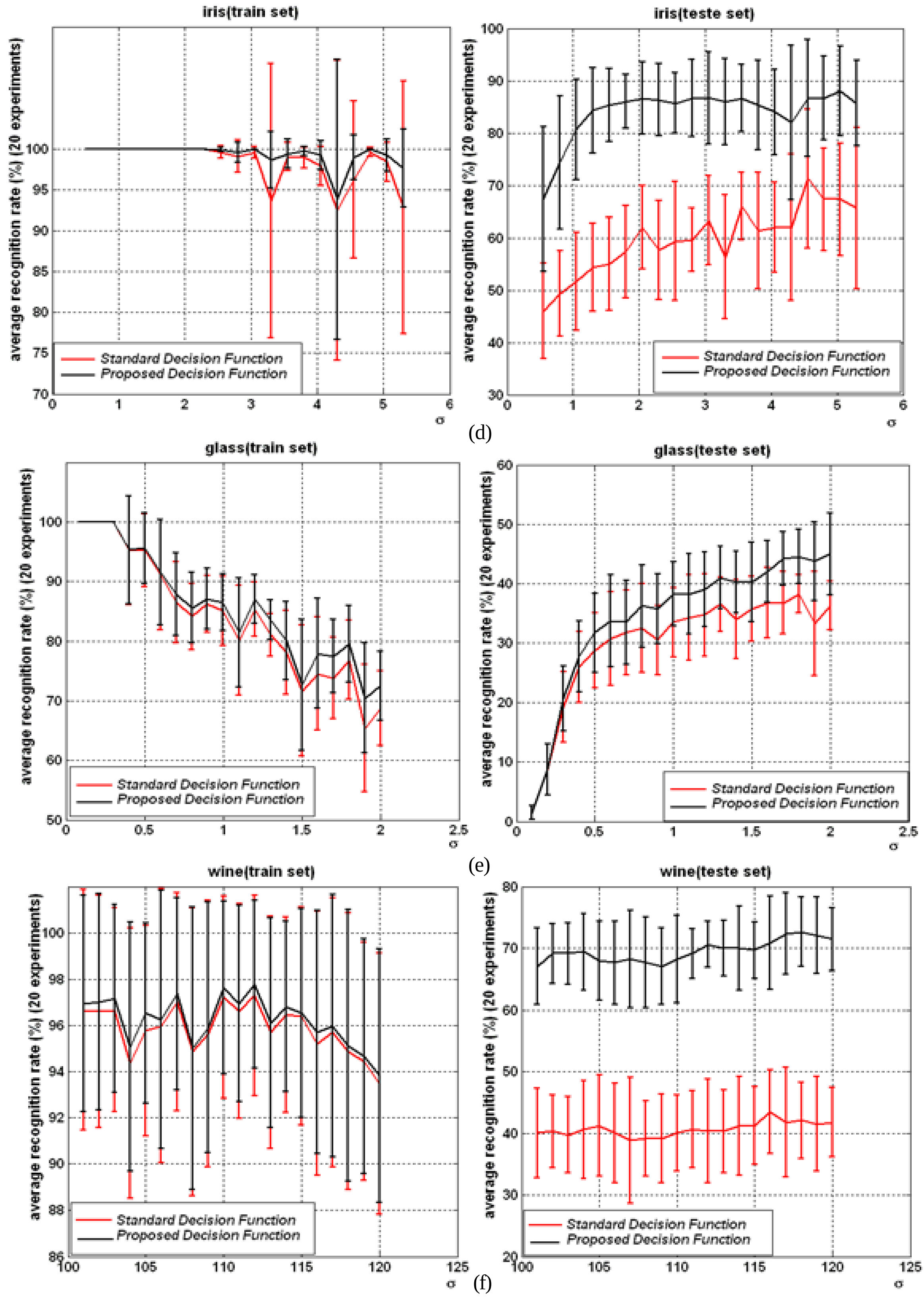


Figure 2. Recognition rate and standard deviation (indicated by symmetric error bars) for train and test set, using (a) monk-1, (b) monk-2, (c) monk-3, (d) iris, (e) wine, (f) glass, against RBF width(σ)

7. Conclusion

In this paper, we proposed a new decision function for support vector data description that resolves, the unclassifiable regions caused by the probable overlap of the hyperspheres, which characterize the classes. In theory, the generalization ability of the new decision function is the same with or better than the standard function, because it react just in the overlapped regions, By computer simulations using six benchmark data sets, we demonstrated the superiority of the proposed method.

References

[1] David M. J. Tax., Robert P. W. Duin. (1999). Data Domain Description Using Support Vectors. *In*: Proceedings European Symposium on Artificial Neural Networks Bruges (Belgium), 251-256

[2] David M. J. Tax, Robert P. W. Duin. (2004). Support Vector Data Description, *Machine Learning*. 54 (1) 45–66.

[3] Asa Ben-Hur, David Horn, Hava T. Siegelmann,Vladimir Vapnik. (2001). Support vector clustering, *Journal of Machine Learning Research*, 2 (12) 125-137.

[4] KiYoung Lee, Dae-Won Kim, Doheon Lee, Kwang H. Lee, (2005). Improving support vector data description using local density degree, *Pattern Recognition*, 38 (10) 1768 – 1771.

[5] Cortes C, Vapnik VN, (1995). Support-vector networks, *Machine Learning*, 20 (3) 273–297.

[6] Http: //www.ics.uci.edu/~ mlearn.

[7] Mingrui Wu and Jieping Ye, (2009). A Small Sphere and Large Margin Approach for Novelty Detection Using Training Data with Outliers. *IEEE Trransactions on Pattern Analysis and Machine Intelligence*, 31 (11) 2088 - 2092.

[8] Ritter, G., Gallegos, M. T. (1997). Outliers in statistical pattern recognition and an application to automatic chromosome classification, *Pattern Recognition Letters*, 18 (6) 525-539.

[9] Xiufeng Jiang , Zhang Yi , Jian Cheng Lv, (2006). Fuzzy SVM with a new fuzzy membership function, *Neural Computing and Application*, 15 (3) 268–276.

[10] Chung, F. L., Wang, S., Deng, Z., Hu, D. (2004). Fuzzy kernel hyperball perceptron, *Applied Soft Computing*, 5 (1) 67–74.

[11] Huang, H. P., Liu, Y. H. (2002). Fuzzy support vector machines for pattern recognition and data mining. *International Journal of Fuzzy Systems*, 4 (3) 826–835.

[12] Jayadeva, K., Suresh C. (2004). Fast and robust learning through fuzzy linear proximal support vector machines, *Neurocomputing*, 61, 401–411.

[13] Chun-Fu Lin, Sheng-De Wang. (2002). Fuzzy support vector machines, *IEEE Transactions on Neural Networks*, 13 (2) 464–471.

[14] Tao, Q., Wang, J. (2004). A new fuzzy support vector machine based on the weighted margin. *Neural Processing Letter*s, 20 (3) 139–150.

[15] Wei, L. L., Long, W. J., Zhang, W. X. (2003). Fuzzy data domain description using support vector machines. *In*: Proceedings of the Second International Conference on Machine Learning and Cybernetics, 5, 3082–3085.

[16] Zheng, C. H., Jiao, L. C, (2002). Fuzzy pre-extracting method for support vector machine. *In*: Proceedings of the first International Conference on Machine Learning and Cybernetics, 4, 2026 - 2030.

[17] Abe, S., Inoue, T. (2002). Fuzzy support vector machines for multiclass problems. *In*: Proceedings of the 10^{th} European symposium on Artificial Neural Networks, Bruges, Belgium., 113–118

[18] Inoue, T., Abe, S. (2001). Fuzzy support vector machines for pattern classification. Proceedings of the International Joint Conference on Neural Networks, 2, 1449–1454.

[19] Kikuchi, T., Abe, S, (2005). Comparison between error correcting output and fuzzy support vector machines. *Pattern Recognition Letters*, 26 (12), 1937–1945.

[20] Takahashi, F., Abe, S. (2002). Decision-tree-based multiclass support vector machines. *In*: Proceedings of the 9^{th} International conference on Neural Information Processing, p. 1418–1422.

[21] Tsujinishi, D., Abe, S. (2003). Fuzzy least squares support vector machines for multiclass problems, *Neural Networks* 16 (5) 785–792.

[22] Zhang XG. (1999). Using class-center vectors to build support vector machines. *In*: IEEE Proceedings of the Neural networks and Signal Processing IX, 3-11.