Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2013, Article ID 928054, 7 pages
http://dx.doi.org/10.1155/2013/928054

Research Article

A Support Vector Classifier Based on Vague Similarity Measure

Yong Zhang and Jing Cai

School of Computer and Information Technology, Liaoning Normal University, No. 1, Liushu South Street, Ganjingzi District, Dalian, Liaoning Province 116081, China

Correspondence should be addressed to Yong Zhang; zhyong@lnnu.edu.cn

Received 11 December 2012; Revised 19 February 2013; Accepted 19 February 2013

Academic Editor: Wei-Chiang Hong

Support vector machine (SVM) is a popular machine learning method for its high generalizaiton ability. How to find the adaptive kernel function is a key problem to SVM from theory to practical applications. This paper proposes a support vector classifer based on vague sigmoid kernel and its similarity measure. The proposed method uses the characteristic of vague set, and replaces the traditional inner product with vague similarity measure between training samples. The experimental results show that the proposed method can reduce the CPU time and maintain the classification accuracy.

1. Introduction

Support vector machine (SVM) constructs a hyperplane or set of hyperplanes in a high-dimensional feature space, which can be used for classification, regression, or other tasks. It was first introduced by Vapnik [1] for classification and has been applied to many application fields successfully. SVM is based on the structural risk minimization principle, which incorporates capacity control to prevent overfitting and thus is a partial solution to the bias-variance trade-off dilemma. The basic idea of SVM classification is to find such a separating hyperplane that corresponds to the largest possible margin between points of different classes.

How to find the adaptive kernel function is a key problem to SVM from theory to practical applications. There are several kernel functions, for example, the linear kernel, the polynomial kernel, and the RBF kernel. These kernel functions are positive semidefinite (PSD). However, some non-PSD matrices are used in practice. An important one is the sigmoid kernel $K(x_i, x_j) = \tanh(ax_i^T x_j + r)$, which is related to neural networks. It was first pointed out by Vapnik [1] that sigmoid kernel matrix might not be PSD for the certain values of parameters, a and r. However, the sigmoid kernel matrix is conditionally positive definite in certain parameters and thus it is valid kernel.

Meanwhile, datasets in real life are usually noisy, and a classifier which is obtained by training with noisy data cannot classify some data samples correctly. So, fuzzy theory is introduced in support vector machines by many researchers in order to solve the above problem. There exist two cases. The first one is fuzzy support vector machine (FSVM) [2–4]. FSVM takes into account the noisy in the training set and associates a fuzzy membership with every sample which will account for the uncertainty in the class to which it belongs. It uses the membership function to express the membership grade of a sample belonging to positive class or negative class. The second one is to combine fuzzy theory to kernel functions of SVM and propose a fuzzy kernel-based SVM. Fuzzy kernel is apt to construct a robust classifier and process classification and regression of uncertain or fuzzy data. Soria-Olivas et al. [5] propose a fuzzy-based activation function for artificial neural networks. Camps-Valls et al. [6] extend the fuzzy-based activation function and propose a support vector classifier based on fuzzy sigmoid kernel. The fuzzy sigmoid function allows lower computational cost and higher rate of positive eigenvalues of the kernel matrix than those from the standard sigmoid kernel [6]. Yang et al. [7] develop a kernel fuzzy c-means clustering-based fuzzy SVM algorithm to deal with the classification problems with outliers or noises.

In fuzzy theories, the vague set theory [8] is one of the methods used to deal with uncertain information and has gradually become more and more popular for handling decision-making problems. Since vague sets can provide more information than fuzzy sets, they are considered superior in mathematical analysis of uncertain information. This paper combines vague sets with sigmoid kernel and proposes a novel support vector classifier based on vague sigmoid kernel.

The rest of this paper is organized as follows: Section 2 reviews the related research and briefly describes the vague set theory and support vector machine. We present a novel support vector classifier based on vague sigmoid kernel and its similarity measure in Section 3. Section 4 presents the experimental results obtained on benchmark data sets and analyzes the performance of the proposed algorithm. Section 5 concludes the paper with some final remarks.

2. Vague Set and Support Vector Machine

2.1. The Conception of Vague Set. Fuzzy set theory was first proposed by Zadeh [9]. It is an important mathematical approach to uncertain and fuzzy data analysis and has successfully been applied in the areas of fuzzy control, fuzzy decision making, and so on.

Introduced by Gau and Buehrer [8], vague set is a generalization of the concept of a fuzzy set. Note that the vague set is the same with the intuitionistic fuzzy set in essence according to some research work [10]. The major advantage of vague set over fuzzy set is that the former makes descriptions of the objective world more realistic, practical, and accurate. Presently, many scholars have been interested in the theory and already made further studies. They have been widely applied in medical diagnosis, decision making, pattern recognition, uncertain knowledge acquisition, and so forth [11–14].

Definition 1 (vague sets [8]). Let X be the universe of discourse, $X = \{x_1, x_2, \dots, x_{n-1}, x_n\}$, with a generic element of X denoted by x_i. A vague set A in X is characterized by a truth-membership function t_A and a false-membership function f_A,

$$t_A : X \longrightarrow [0,1], \qquad f_A : X \longrightarrow [0,1], \tag{1}$$

where $t_A(x_i)$ is a lower bound on the grade of membership of x_i derived from the evidence for x_i, $f_A(x_i)$ is a lower bound on the negation of x_i derived from the evidence against x_i, and $t_A(x_i) + f_A(x_i) \leq 1$. It is clear that the grade of membership of x_i in the vague set A has been restricted in a subinterval $[t_A(x_i), 1 - f_A(x_i)]$ of $[0,1]$. The subinterval $[t_A(x_i), 1 - f_A(x_i)]$ is called vague value of x_i in vague set A.

The vague value $[t_A(x_i), 1-f_A(x_i)]$ indicates that the exact grade of membership $\mu_A(x_i)$ of x_i may be unknown but is bounded by $t_A(x_i) \leq \mu_A(x_i) \leq 1 - f_A(x_i)$.

When the universe of discourse X is continuous, a vague set A can be written as

$$A = \int_X \frac{[t_A(x_i), 1 - f_A(x_i)]}{x_i}. \tag{2}$$

When the universe of discourse X is discrete, a vague set A can be written as

$$A = \sum_{i=1}^{n} \frac{[t_A(x_i), 1 - f_A(x_i)]}{x_i}. \tag{3}$$

Let $\pi_A(x_i)$ be an uncertain degree of x_i in vague set A, $\pi_A(x_i) = 1 - t_A(x_i) - f_A(x_i)$. $\pi_A(x_i)$ characterizes the precision of our knowledge about x_i. If $\pi_A(x_i)$ is small, our knowledge about x_i is relatively precise; if it is large, we know correspondingly little. If $1 - f_A(x_i)$ is equal to $t_A(x_i)$, our knowledge about x_i is exact, and the theory reverts back to that of fuzzy sets. If both $1 - f_A(x_i)$ and $t_A(x_i)$ are equal to 1 or 0, our knowledge about x_i is very exact, and the theory reverts back to that of ordinary sets.

For example, let A be a vague set with truth-membership function t_A and false-membership function f_A, respectively. If a vague value is [0.5, 0.8], then according to Definition 1 we can see that $t_A(x_i) = 0.5, f_A(x_i) = 1 - 0.8 = 0.2$, and $\pi_A(x_i) = 1 - 0.5 - 0.2 = 0.3$. It can be interpreted as "assume that the total number of the votes is 10, the votes for a resolution is 5 in favor, 2 against, and 3 abstentions." Obviously, fuzzy set cannot exactly denote and process the type of obscure information.

Many similarity measures have been proposed in the literature for measuring the degree of similarity between vague sets. Chen [15, 16] proposed the concept of similarity measures between vague sets and defined its expression $M_C(X,Y)$ as follows: $M_C(X,Y) = 1 - |(S(X) - S(Y))/2|$, where X and Y are two vague values, $S(X) = t_X - f_X$, and $S(Y) = t_Y - f_Y$. It is obvious that the larger the value of $M_C(X,Y)$, the more the similarity between the vague values X and Y. Hung and Yang [11] presented three new similarity measures between intuitionistic fuzzy sets based on Hausdorff distance. Li et al. [17] analyzed and summarized several similarity measures between vague sets. Dou et al. [18] developed a new similarity measure of vague sets and defined the new relative degree of similarity measures to solve the fuzzy shortest path problem.

2.2. Support Vector Machine and Its Kernel. In this section, we briefly review the learning algorithm of support vector machine (SVM) initially proposed in [1]. Given a binary classification problem represented by a dataset $\{(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)\}$, where $x_i \subset \mathfrak{R}^n$ represents an n-dimensional data sample and $y_i \in \{+1, -1\}$ represents the class of that data sample, for $i = 1, \dots, l$, the goal of the SVM learning algorithm is to find an optimal hyperplane that separates these data samples into two classes. In order to find a better separation of classes, the data are first transformed into a higher-dimensional feature space by a mapping function ϕ. Then, a possible separating hyperplane, which resides in the higher-dimensional feature space, can be represented by

$$w \cdot \phi(x) + b = 0. \tag{4}$$

The support vector technique requires the solution of the following optimization problem:

$$\min_{\mathbf{w},\mathbf{b},\xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{n} \xi_i \tag{5}$$

subject to the constraints

$$y_i (w \cdot \phi(x_i) + b) + \xi_i - 1 \geq 0 \quad \xi_i \geq 0, \ i = 1, \dots, n, \tag{6}$$

where the training vectors x_i are mapped into a higher-dimensional space by the function ϕ parameter C is a user-specified positive parameter that controls the trade-off between maximizing the margin and minimizing the training error term. The slack variables $\xi_i > 0$ hold for misclassified samples, and therefore, $\sum_{i=1}^{l} \xi_i$ can be thought of as a measure of the amount of misclassifications. This quadratic optimization problem can be solved by constructing a Lagrangian representation and transforming it into the following dual problem:

$$\begin{aligned} \max_{\lambda} W(\lambda) &= \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j x_i \cdot x_j \\ &= \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j K(x_i, x_j) \end{aligned} \tag{7}$$

subject to the constraints

$$\sum_{i=1}^{n} \lambda_i y_i = 0, \quad 0 \le \lambda_i \le C, \ i = 1, \ldots, n, \tag{8}$$

where λ_i is the Lagrangian parameter. Note that the kernel trick $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ is used in the last equality in (7). The Karush-Kuhn-Tucker conditions of SVM are defined by

$$\begin{aligned} \lambda_i [y_i (w \cdot \phi(x_i) + b) - 1 + \xi_i] = 0, \quad i = 1, \ldots, n, \\ (C - \lambda_i) \xi_i = 0, \quad i = 1, \ldots, n. \end{aligned} \tag{9}$$

The sample x_i with the corresponding nonzero λ_i is called a support vector. The optimal value of weight vector w_0 is obtained by $w_0 = \sum_{i=1}^{n} \lambda_i y_i \phi(x_i) = \sum_{i=1}^{n_s} \lambda_i y_i \phi(x_i)$, where n_s is the number of support vectors. The optimal value of bias b_0 can be computed from the Karush-Kuhn-Tucker conditions (9); namely, $b_0 = y_i - w_0 \cdot \phi(x_i)$, for a random support vector sample x_i. Once the optimal pair (w_0, b_0) is determined, the SVM decision function is then given by

$$f(x) = \operatorname{sign}\left(\sum_{i=1}^{n_s} \lambda_i y_i K(x, x_i) + b \right), \tag{10}$$

where $K(x_i, x_j)$ is called the kernel function as follows:

$$K(x_i, x_j) = \phi(x_i)^T \phi(x_j). \tag{11}$$

Several typical kernel functions are the linear kernel $K(x_i, x_j) = x_i \cdot x_j$, the polynomial kernel $K(x_i, x_j) = (a x_i^T x_j + r)^d$, and the RBF kernel $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$.

The kernel functions above must satisfy Mercer condition. Namely, the kernel function matrix is a symmetric and positive semidefinite (PSD) matrix. Nevertheless, some non-PSD matrices are used in SVM in practice [19]. The sigmoid kernel $K(x_i, x_j) = \tanh(a x_i^T x_j + r)$ is an available non-PSD kernel function. The sigmoid kernel is also known as the hyperbolic tangent kernel and as the multilayer perceptron (MLP) kernel, which comes from the neural networks field.

It was first pointed out by Vapnik [1] that its kernel matrix might be non-PSD for certain values of the paramenters α and r. When the kernel function is non-PSD, (11) cannot be satisfied. H. T. Lin and C. J. Lin [19] also study non-PSD kernel function and its applications to SVM and testify the sigmoid kernel matrix is conditionally positive definite (CPD). When parameters $a > 0$ and $r < 0$, the sigmoid kernel is suitable for a valid kernel. The sigmoid kernel has been used in several practical cases, such as support vector machine classification [6, 19], decision rules extraction [20], and chaotic time series prediction [21].

3. The Proposed Support Vector Classifier Based on Vague Similarity Measure

This section presents the proposed method for vague sigmoid kernel-based support vector classifier. It first gives a brief introduction to fuzzy kernel and then focuses on a proposed algorithm.

3.1. Fuzzy Kernel. Several researchers have studied fuzzy kernel. Kwan [22] proposes a simple sigmoid-like nonlinear activation function more suitable for digital hardware implementation as follows:

$$f(x) = \begin{cases} \operatorname{sign}(x), & |x| \ge L \\ -\dfrac{x \cdot |x|}{L^2} + \dfrac{2x}{L}, & \text{otherwise}, \end{cases} \tag{12}$$

where L is the width of the transition region.

Inspired by the work of Kwan, Soria-Olivas et al. [5] think that the activation function of (12) can be drawn in a more natural way by defining the classical activation function by means of the fuzzy logic methodology and propose a fuzzy-based activation function for artificial neural networks, which considers triangular functions due to their simplicity. The fuzzy-based sigmoid function models the hyperbolic tangent function by means of linguistic variables. Camps-Valls et al. [6] extend the fuzzy-based activation function and propose a support vector classifier based on fuzzy sigmoid kernel. The fuzzy sigmoid kernel allows lower computational cost and higher rate of positive eigenvalues of the kernel matrix, which alleviates current limitations of the sigmoid kernel.

Although fuzzy set theory can preferably characterize fuzziness, it has an obvious shortage due to using a single-value membership $\mu_A(u_i) \in [0, 1]$ to represent the degree of membership. Fuzzy theory lacks consideration of some nondeterministic factors among samples. In this case, we propose a fuzzy sigmoid kernel support vector classifier based on vague theory.

3.2. Vague Value Computation. The proposed algorithm considers two-class classification for simplicity. We first decide the class center of each class, namely, x_A^* and x_B^* in Figures 1–3, and then compute vague values of samples.

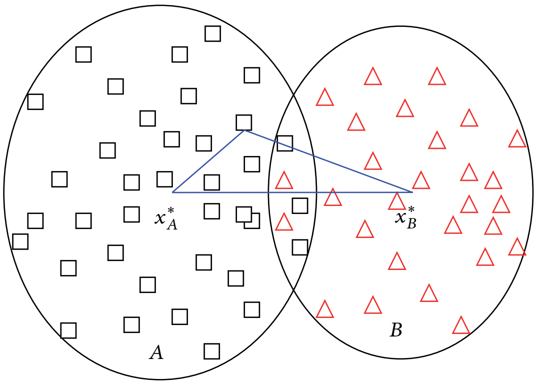


Figure 1: Vague value computation in class A.

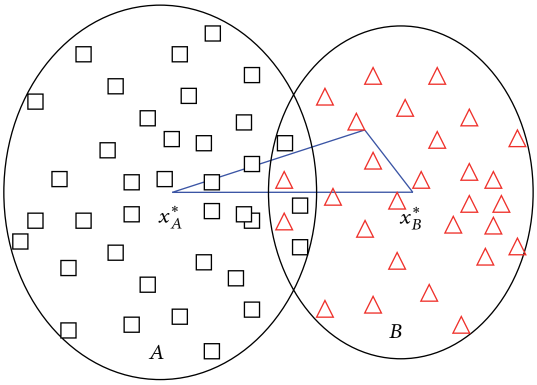


Figure 2: Vague value computation in class B.

For the sample x in the training set, if x belongs to class A but does not belong to the intersection area between class A and class B, we define vague value of x as follows:

$$v(x) = \left[1 - \frac{\|x - x_A^*\|}{r_A},\ 1 - \frac{\|x - x_A^*\|}{r_A}\right], \tag{13}$$

where r_A is the radius of class A. This case is shown in Figure 1. Similarly, let r_B be the radius of class B, we get $\|x - x_B^*\|/r_B > 1$.

If x belongs to class B but does not belong to the intersection area between class A and class B, we define vague value of x as follows:

$$v(x) = \left[1 - \frac{\|x - x_B^*\|}{r_B},\ 1 - \frac{\|x - x_B^*\|}{r_B}\right], \tag{14}$$

where r_B is the radius of class B. This case is shown in Figure 2. We get $\|x - x_A^*\|/r_A > 1$.

If sample x belongs to the intersection area, we label m_A and m_B as $m_A = \|x - x_A^*\|/r_A$ and $m_B = \|x - x_B^*\|/r_B$ and define vague value of x as follows:

$$v(x) = \begin{cases} [m_A(1 - m_B), m_B(1 - m_A)], & \text{if } m_A \le m_B \\ [m_B(1 - m_A), m_A(1 - m_B)], & \text{if } m_A > m_B. \end{cases} \tag{15}$$

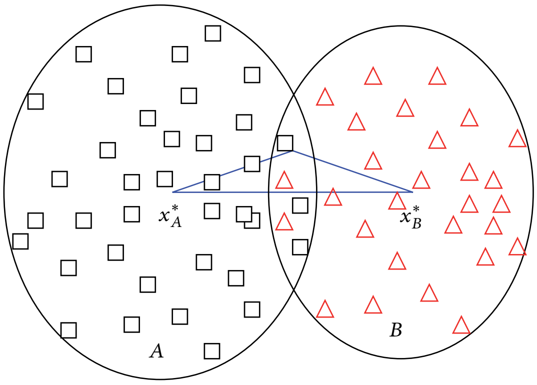


Figure 3: Vague value computation in intersection area between class A and class B.

This case is shown in Figure 3.

Through a detailed analysis of samples in class A and class B, we find $t_A(x) = 1 - f_A(x)$ shown in Figures 1 and 2, and vague set reverts back to fuzzy set in two cases. For sample x in Figure 3, we reduce the classification effect of these samples.

3.3. Similarity Measure of Vague Value and Its Kernel. For samples x_i and x_j, their inner $x_i \cdot x_j$ is obtained typically by computing Euclidean distance between two samples. In this paper, we replace Euclidean distance with similarity measures between vague sets after introducing vague membership. A similarity measure is used for estimating the degree of similarity between two sets. The main idea is described below. We first compute vague values of samples and then represent these vague values with points in the spatial coordinate system. At last, we compute similarity measures between points.

Definition 2 (see [23]). Let $v(x_i)$ be the vague value of sample x_i computed by the above method. The corresponding point in the spatial coordinate system is represented as $D_i(t(x_i)(1 + \pi(x_i)), f(x_i)(1 + \pi(x_i)), \pi^2(x_i))$. We also denote a 3-tuple $D_i(T(x_i), F(x_i), \Pi(x_i))$ for simplicity, where $T(x_i) = t(x_i)(1 + \pi(x_i))$, $F(x_i) = f(x_i)(1 + \pi(x_i))$, and $\Pi(x_i) = \pi^2(x_i)$.

As shown in Definition 2, D_i in the spatial coordinate system includes three parts, $t(x_i)$, $f(x_i)$, and $\pi(x_i)$, respectively. The meanings of $t(x_i)$, $f(x_i)$, and $\pi(x_i)$ are shown in Definitions 1 and 2. Analyzing from the vote model, we consider that some abstentions are likely prone to be in favor, others are likely prone to be against, and others are likely to be abstention. So, we further divide abstention part into three parts: $t(x_i)\pi(x_i)$, $f(x_i)\pi(x_i)$, and $\pi^2(x_i)$, which represent the cases of being in favor, against, and abstention in all abstentions, respectively. We can use a point in three-dimensional space to depict a membership degree of a training sample.

Obviously, $t(x_i) + f(x_i) + \pi(x_i) = 1$. Using Definition 3, we can get a point in three-dimensional space D_i. For three parts

Table 1: Characteristics of experimental data sets.

Dataset	No. of samples	No. of attributes	No. of classes
Ionosphere	351	34	2
Sonar	208	60	2
Pima-diabetes	768	8	2
Wdbc	569	30	2
Iris	150	43	3
Vehicle	846	18	4

in D_i, $t(x_i)(1 + \pi(x_i)) + f(x_i)(1 + \pi(x_i)) + \pi^2(x_i) = (t(x_i) + f(x_i))(1+\pi(x_i)) + \pi^2(x_i) = (1-\pi(x_i))(1+\pi(x_i)) + \pi^2(x_i) = 1$.

Definition 3. Let D_i and D_j be two points defined as Definition 2; then their similarity measures can be defined as follows:

$$\begin{aligned} & T\left(D_i, D_j\right) \\ & = \left\| D_i - D_j \right\| \\ & = \sqrt{\left(T\left(x_i\right) - T\left(x_j\right)\right)^2 + \left(F\left(x_i\right) - F\left(x_j\right)\right)^2 + \left(\Pi\left(x_i\right) - \Pi\left(x_j\right)\right)^2}. \end{aligned} \tag{16}$$

Based on vague value and similarity measure above, we give a computation method of vague sigmoid kernel function. Expression (12) can be readily rewritten as a function of a and r as follows:

$$K\left(x_i, x_j\right) = \begin{cases} \operatorname{sign}(M), & |M| \geq \dfrac{1}{a} \\ 2\alpha M - \alpha^2 M \cdot |M|, & \text{otherwise,} \end{cases} \tag{17}$$

where $M = T(D_i, D_j) + r/a$.

3.4. The Proposed Vague Sigmoid-Based Support Vector Classifier. In order to compute vague values of training samples, we first decide the class center of each class. In this paper, we select fuzzy c-means (FCM) method [24, 25] to do it. In fuzzy clustering, FCM method has become one of the most popular techniques.

FCM algorithm starts with an initial guess for the cluster centers, which intends to mark the mean location of each cluster [24, 25]. 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,

$$\min \quad J(U, V) = \sum_{i=1}^{c} \sum_{k=1}^{l} u_{ik}^{m} d\left(x_k, v_i\right), \tag{18}$$

$$\text{s.t.} \quad \sum_{i=1}^{c} u_{ik} = 1, \quad 0 \leq u_{ik} \leq 1, \tag{19}$$

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, $d(x_k, v_i) = \|x_k - v_i\|^2$ denotes the distance between point x_k and the cluster center v_i, and V is the set of cluster centers or prototypes ($v_i \in R^p$). When the objects change clusters, the membership values are recalculated according to the following formula:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} \left(d\left(x_k, p_i\right) / d\left(x_k, v_j\right)\right)^{2/(m-1)}}. \tag{20}$$

Each cluster center is then calculated by

$$v_i = \frac{\sum_{k=1}^{l} \left(u_{ik}\right)^m x_k}{\sum_{k=1}^{l} \left(u_{ik}\right)^m}. \tag{21}$$

After getting the cluster centers, the algorithm can compute vague values of training sample and measure the similarity of vague values.

According to analysis above, we propose a novel vague sigmoid-based support vector classifier. The algorithm description is as follows.

Step 1. Preprocess data set and classify training data set and testing data set.

Step 2. Use FCM algorithm to compute the membership u_{ik} of each training sample x_i to each class k and to obtain the cluster centers.
Substep 2.1. Select the number of clusters c, the maximal iterative count $N_{\max}$, fuzziness parameter m (let $m = 2$) and converge error $\varepsilon > 0$.
Substep 2.2. Initialize the membership matrix $U^{(0)} = \{u_{ik}^{(0)}\}$ satisfied with constraint conditions

$$\sum_{i=1}^{c} u_{ik} = 1, \quad 0 \leq u_{ik} \leq 1. \tag{22}$$

Substep 2.3. For $t = 1, \ldots, N_{\max}$,

calculate the membership matrix $U^{(t)} = \{u_{ik}^{(t)}\}$ according to (20);

calculate the cluster centers $v_i^{(t)}$ $(i = 1, \ldots, c)$ according to (21);

calculate the objective function $J^{(t)}(U, V)$ according to (18);

Table 2: Accuracy (%) and time (s) comparison of different methods.

Dataset	Sigmoid		Fuzzy sigmoid		Vague sigmoid	
	Accuracy (%)	Time (s)	Accuracy (%)	Time (s)	Accuracy (%)	Time (s)
Ionosphere	94.8	22.30	93.4	20.52	94.3	21.83
Sonar	88.2	31.85	88.1	24.87	87.9	28.32
Pima-diabetes	70.9	17.56	72.8	12.38	73.5	14.54
Wdbc	97.3	21.12	97.9	18.04	97.9	19.28
Iris	97.4	4.35	93.8	3.87	96.1	4.21
Vehicle	71.2	36.74	65.3	23.47	70.8	30.42

when $|J^{(t)}(U,V) - J^{(t-1)}(U,V)| < \varepsilon$ or $t = N_{\max}$, stop iteration and return the membership matrix $U^{(t)}$ and the cluster centers $v_i^{(t)}$.

Step 3. Compute vague values of training samples using (13)–(15).

Step 4. Use SVM based on vague sigmoid kernel to train samples with vague values.

The key steps of the proposed algorithm are to compute vague values and to compute vague sigmoid kernel $K(x_i, x_j) = \tanh(a x_i^T x_j + r)$.

4. Experimental Analysis

Six data sets from the University of California at Irvine (UCI) machine learning repository [26] are used in our experiments. The data sets include *Ionosphere, Sonar, Pima-diabetes, Wdbc, Iris and Vehicle*. *Iris,* and *Vehicle* data are multiclass problems. The other data are binary classification problems. The characteristics of these data sets are shown in Table 1. We tested the proposed vague sigmoid kernel method and compared it to sigmoid-based SVM and fuzzy sigmoid-based SVM. The results of these methods depend on the values of the kernel parameters a and r and penalization parameter C.

In order to test the proposed methods, SVM models were trained by using LIBSVM [27]. Parameter a was fixed to $1/A$, where A is the input dimension of data set, and other parameters of all methods were optimized using grid-based 5-fold cross-validation method. For all the datasets each training set and each testing set were the same for all methods. For all the datasets, we used the 5-fold cross-validation method to estimate the accuracy of the classifiers. We compare classification accuracy and CPU time (s) for the sigmoid, the fuzzy-based sigmoid, and the vague-based sigmoid kernels. Experimental results are shown in Table 2.

From Table 2, we can find that the sigmoid-based method achieves the better accuracy than the fuzzy sigmoid method and the vague sigmoid method on *Ionosphere, Sonar, Iris, and Vehicle* data sets. As we can see, the accuracy of the fuzzy sigmoid method and the vague sigmoid method is better than that of the sigmoid-based method on *Pima-diabetes* and *Wdbc* data sets. However, the sigmoid-based method has also more CPU times than the other methods. Notice that the vague sigmoid method achieves the same or slightly better accuracy than the fuzzy sigmoid method.

If we consider the cost of accuracy and CPU time, we may prefer the solution found with the vague sigmoid method. We can find that the average CPU time in the all 6 data sets is 22.32 s in the sigmoid method, while there is 19.77 s in the vague sigmoid method; in the meantime, the classification accuracy does not decrease remarkably.

5. Conclusions

Support vector machine is a novel machine learning method which has been applied to many application fields successfully. The sigmoid kernel was quite popular for support vector machines due to its origin from neural networks. In this paper, we propose a vague sigmoid kernel-based support vector classifier. The proposed method is combined with vague set methodology, which makes the computation of SVM simple. In vague sigmoid kernel, we replace the inner product computation using Euclidean distance between two samples with similarity measures. The experiment is conducted by using 6 data sets from the UCI machine learning repository. The results of classification are evaluated and compared in terms of the performance using accuracy and time. The results obtained from the experiment indicated that the proposed method can reduce the CPU time and maintain the classification accuracy.

Acknowledgments

This work is supported by China Postdoctoral Science Foundation (no. 20110491530), the Science Research Plan of Liaoning Education Bureau (no. L2011186), and the Dalian Science and Technology Planning Project of China (no. 2010J21DW019).

References

[1] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.

[2] J. H. Chen and C. S. Chen, “Fuzzy kernel perceptron,” *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1364–1373, 2002.

[3] D. Tsujinishi and S. Abe, “Fuzzy least squares support vector machines for multiclass problems,” *Neural Networks*, vol. 16, no. 5-6, pp. 785–792, 2003.

[4] C. F. Lin and S. D. Wang, "Training algorithms for fuzzy support vector machines with noisy data," *Pattern Recognition Letters*, vol. 25, no. 14, pp. 1647–1656, 2004.

[5] E. Soria-Olivas, J. D. Martín-Guerrero, G. Camps-Valls, A. J. Serrano-López, J. Calpe-Maravilla, and L. Gómez-Chova, "A low-complexity fuzzy activation function for artificial neural networks," *IEEE Transactions on Neural Networks*, vol. 14, no. 6, pp. 1576–1579, 2003.

[6] G. Camps-Valls, J. D. Martín-Guerrero, J. L. Rojo-Álvarez, and E. Soria-Olivas, "Fuzzy sigmoid kernel for support vector classifiers," *Neurocomputing*, vol. 62, no. 1–4, pp. 501–506, 2004.

[7] X. Yang, G. Zhang, J. Lu, and J. Ma, "A kernel fuzzy c-means clustering-based fuzzy support vector machine algorithm for classification problems with outliers or noises," *IEEE Transactions on Fuzzy Systems*, vol. 19, no. 1, pp. 105–115, 2011.

[8] W. L. Gau and D. J. Buehrer, "Vague sets," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 2, pp. 610–614, 1993.

[9] L. A. Zadeh, "Fuzzy sets," *Information and Computation*, vol. 8, pp. 338–353, 1965.

[10] H. Bustince and P. Burillo, "Vague sets are intuitionistic fuzzy sets," *Fuzzy Sets and Systems*, vol. 79, no. 3, pp. 403–405, 1996.

[11] W. L. Hung and M. S. Yang, "Similarity measures of intuitionistic fuzzy sets based on Hausdorff distance," *Pattern Recognition Letters*, vol. 25, no. 14, pp. 1603–1611, 2004.

[12] D. Zhang, J. Zhang, K. K. Lai, and Y. Lu, "An novel approach to supplier selection based on vague sets group decision," *Expert Systems with Applications*, vol. 36, no. 5, pp. 9557–9563, 2009.

[13] J. Ye, "Using an improved measure function of vague sets for multicriteria fuzzy decision-making," *Expert Systems with Applications*, vol. 37, no. 6, pp. 4706–4709, 2010.

[14] L. Feng, T. Li, D. Ruan, and S. Gou, "A vague-rough set approach for uncertain knowledge acquisition," *Knowledge-Based Systems*, vol. 24, no. 6, pp. 837–843, 2011.

[15] S. M. Chen, "Measures of similarity between vague sets," *Fuzzy Sets and Systems*, vol. 74, no. 2, pp. 217–223, 1995.

[16] S. M. Chen, "Similarity measures between vague sets and between elements," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 27, no. 1, pp. 153–158, 1997.

[17] Y. Li, D. L. Olson, and Z. Qin, "Similarity measures between intuitionistic fuzzy (vague) sets: a comparative analysis," *Pattern Recognition Letters*, vol. 28, no. 2, pp. 278–285, 2007.

[18] Y. Dou, L. Zhu, and H. S. Wang, "Solving the fuzzy shortest path problem using multi-criteria decision method based on vague similarity measure," *Applied Soft Computing*, vol. 12, no. 6, pp. 1621–1631, 2012.

[19] H. T. Lin and C. J. Lin, "A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods," Tech. Rep., Department of Computer Science and Information Engineering, National Taiwan University, 2003, http://www.csie.ntu.edu.tw/~cjlin/papers/tanh.pdf.

[20] Q. Liu, Y. Zhang, and Z. Hu, "Extracting decision rules from sigmoid kernel," in *Proceedings of the 4th International Conference on Advanced Data Mining and Applications*, pp. 294–304, 2008.

[21] H. Liu, D. Liu, and L. F. Deng, "Chaotic time series prediction using fuzzy sigmoid kernel-based support vector machines," *Chinese Physics*, vol. 15, no. 6, pp. 1196–1200, 2006.

[22] H. K. Kwan, "Simple sigmoid-like activation function suitable for digital hardware implementation," *Electronics Letters*, vol. 28, no. 15, pp. 1379–1380, 1992.

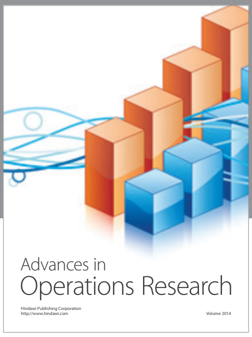
[23] H. W. Liu, "Basis of fuzzy pattern recognition-similarity measures," *Pattern Recognition and Artificial Intelligence*, vol. 17, no. 2, pp. 141–145, 2004 (Chinese).

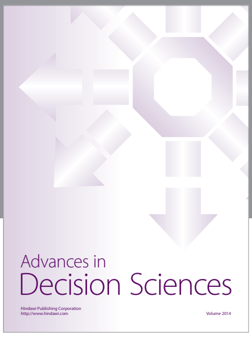
[24] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," *Journal of Cybernetics*, vol. 3, no. 3, pp. 32–57, 1973.

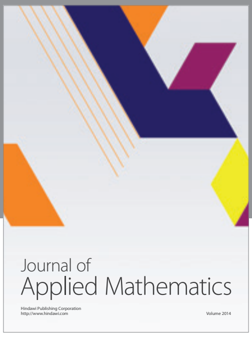
[25] J. C. Bezdek, "A convergence theorem for the fuzzy ISODATA clustering algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, no. 1, pp. 1–8, 1980.

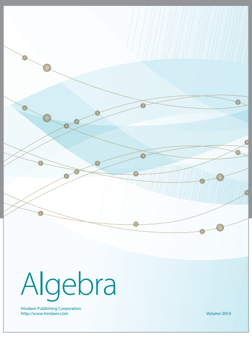
[26] A. Frank and A. Asuncion, *UCI Machine Learning Repository*, School of Information and Computer Science, University of California, Irvine, Calif, USA, 2010, http://archive.ics.uci.edu/ml/.

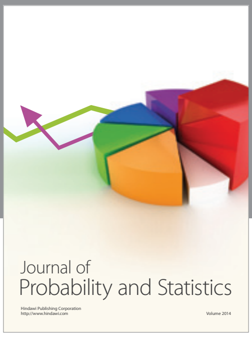
[27] C. C. Chang and C. J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, article 27, 2011.


Advances in
Operations Research

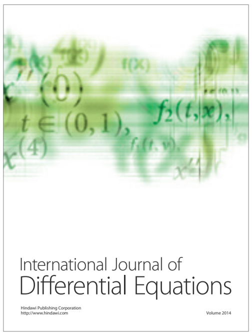

Advances in
Decision Sciences


Journal of
Applied Mathematics

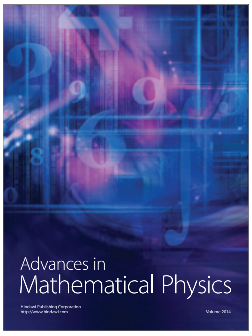

Algebra

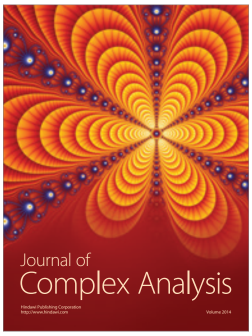

Journal of
Probability and Statistics

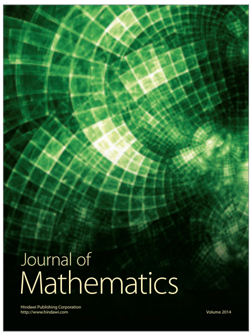
The Scientific
World Journal

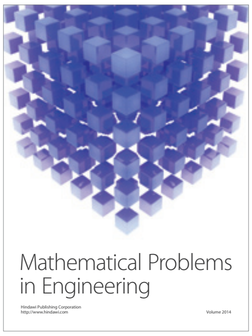

International Journal of
Differential Equations

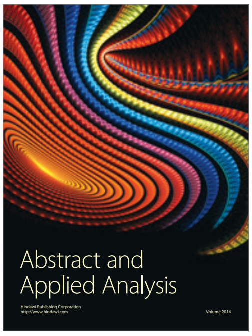
International Journal of
Combinatorics

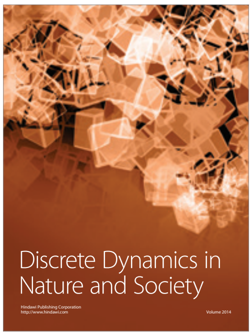

Advances in
Mathematical Physics

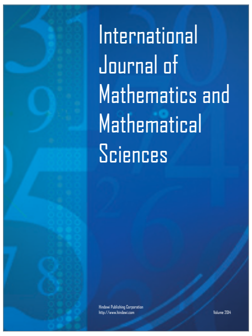

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

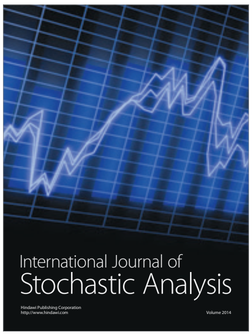

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization