Weighted Twin Support Vector Machine with Universum

Shuxia Lu[1,*], Le Tong[2]

[1] College of Mathematics and Computer Science, Hebei University

Baoding, Hebei, 071002,China

cmclusx@126.com

[2] College of Mathematics and Computer Science, Hebei University

Baoding, Hebei, 071002,China

tonglmo@163.com

Abstract

Universum is a new concept proposed recently, which is defined to be the sample that does not belong to any classes concerned. Support Vector Machine with Universum ($\mathcal{U}$-SVM) is a new algorithm, which can exploit Universum samples to improve the classification performance of SVM. In fact, samples in the different positions have different effects on the bound function. Then, we propose a weighted Twin Support Vector Machine with Universum (called $\mathcal{U}$-WTSVM), where samples in the different positions are proposed to give different penalties. Therefore $\mathcal{U}$-WTSVM has better flexibility of the algorithm and can obtain more reasonable classifier in most case. All experiments demonstrate that our $\mathcal{U}$-WTSVM far outperforms TSVM, and lightly outperforms $\mathcal{U}$-TSVM not only in linear case but also in nonlinear case.

Keywords: Universum, TSVM, $\mathcal{U}$-SVM, Weight.

1. Introduction

SVM with Universum is a new method and the basic idea is to incorporate a priori knowledge into the learning process. The Universum sample concept has been introduced by Weston, Collobert, Sinz, Bottou, and Vapnik in 2006 [1]. The Universum is defined as a collection of unlabeled examples known not belong to any class. Then Weston et al., give a modified Support Vector Machine (SVM) [12] framework, called $\mathcal{U}$-SVM [1] and their experiment results express that $\mathcal{U}$-SVM outperforms those SVMs without considering Universum data. Sinz et al., gave an analysis of $\mathcal{U}$-SVM [2]. Examples from the universum are not belonging to any of the classes the learning task concerns, but they reflect a priori knowledge about application domain [1]. About the Universum data improvement can be found in [4] .Some extensions to the $\mathcal{U}$-SVM can be found in [3]-[7].

In order to improve the SVM computational speed, Jayadeva et al. [8] proposed a twin support vector machine (TSVM) classifier for binary classification. TSVM generates two nonparallel hyper-planes by solving two small QPPs such that each hyperplane is closer to one class and is as far as possible from the other one [9]. Solving two smaller sized QPPs rather than a single large one, makes the learning speed of TSVM approximately four times faster than the standard SVM. Other literatures also can be found in [10] and [11].

In this paper, inspired by the success of [1] and [12], we proposed a weighted Twin Support Vector Machine with Universum (called $\mathcal{U}$-WTSVM), give different penalties to the samples depending their different positions. This method not only retains the superior characteristics of $\mathcal{U}$-TSVM, but also has its additional advantages: comparable or better classification accuracy compared to $\mathcal{U}$-SVM, and TSVM. Moreover, the running time does not increase much.

The remaining parts of the paper are organized as follows. Section 2 briefly introduces the background of TSVM and $\mathcal{U}$-SVM; Section 3 describe the detail of $\mathcal{U}$-WTSVM; All public datasets experiment results are shown in the section 4; In the last section gives the conclusions.

2. Background

In this section, we give a brief outline of TSVM[8] and $\mathcal{U}$-SVM[1]. For classification about the training set $T = \{(x_1, y_1), (x_2, y_2), \dots (x_m, y_m)\}$, where $x_i \in R^n$ and $y_i \in \{1, -1\}, i = 1, \dots, m$.

2.1 TSVM

Consider a binary classification problem of m_1 positive points and m_2 negative points. Suppose that the data points in class +1 are denoted by $A \in R^{m_1 \times n}$, where each row $A_i \in R^n$ represents a data point. Similarly, $B \in R^{m_2 \times n}$ represents the data points of class -1.

ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 2, No.8 , March 2014
ISSN : 2322-5157
www.ACSIJ.org

Linear TSVM searches for two nonparallel hyperplanes

$$f_1(x) = \omega_1^T x + b_1 = 0$$

$$f_2(x) = \omega_2^T x + b_2 = 0$$

(1)

where $\omega_1 \in R^n$, $\omega_2 \in R^n$, $b_1 \in R$ and $b_2 \in R$. Such that each hyperplane is proximal to the data points of one class and far from the other class.

The TSVM classifier is obtained by solving the following pair of QPPs

$$\min_{\omega_1,b_1,\xi} \frac{1}{2}(A\omega_1 + e_1 b_1)^T (A\omega_1 + e_1 b_1) + c_1 e_2^T \xi$$

$$s.t. \; -(B\omega_1 + e_2 b_1) + \xi \geq e_2, \; \xi \geq 0,$$

(2)

And

$$\min_{\omega_2,b_2,\eta} \frac{1}{2}(A\omega_2 + e_2 b_2)^T (A\omega_2 + e_2 b_2) + c_2 e_1^T \eta$$

$$s.t. \; -(A\omega_2 + e_1 b_2) + \eta \geq e_1, \; \eta \geq 0,$$

(3)

where $c_1, c_2 \geq 0$ are parameters and e_1, e_2 are vectors of one's of appropriate dimensions. By introducing the Lagrangian multipliers, we obtain the Wolfe dual of TSVM as follows

$$\max_{\alpha} \; e_2^T \alpha - \frac{1}{2}\alpha^T G(H^T H)^{-1} G^T \alpha$$

$$s.t. \quad 0 \leq \alpha \leq c_1$$

(4)

and

$$\max_{\gamma} \; e_1^T \gamma - \frac{1}{2}\gamma^T P(Q^T Q)^{-1} P^T \gamma$$

$$s.t. \quad 0 \leq \gamma \leq c_2$$

(5)

Here, $G = [B \; e_2]$, $H = [A \; e_1]$, $P = [A \; e_1]$, and $Q = [B \; e_2]$, and the nonparallel proximal hyperplanes are obtained from the solution α and γ of (4) and (5) by

$$v_1 = -(H^T H)^{-1} G^T \alpha \;\; \text{and} \;\; v_2 = -(Q^T Q)^{-1} P^T \gamma$$

Where $v_1 = [\omega_1, b_1]^T$, $v_2 = [\omega_2, b_2]^T$. The case of nonlinear kernels is handled on lines similar to linear case.

2.2 U-SVM

The basic theory of Universum SVM is based on the standard SVM method, by adding unlabeled samples in the training process in order to get more information on the distribution of samples.

Let $L = \{(x_1, y_1), \dots (x_m, y_m)\}$ be the set of labeled examples and let $U = \{z_1, \dots, z_u\}$ denote the set of Universum examples. A standard SVM using the hinge loss $H_a[t] = max\{0, a - t\}$ can compactly be formulated as

$$\min_{\omega,b} \; \frac{1}{2}\|\omega\|^2 + C_L \sum_{i=1}^{m} H[y_i f_{\omega,b}(x_i)]$$

(6)

the prior knowledge embedded in the Universum can be reflected in the sum of the losses $\sum_{j=1}^{u} I[f_{\omega,b}(z_j)]$, U-SVM use the ϵ-insensitive loss $I[t] = H_{-\epsilon}[t] + H_{-\epsilon}[-t]$ for Universum

$$\min_{\omega,b} \; \frac{1}{2}\|\omega\|_2^2 + C_L \sum_{i=1}^{m} H[y_i f_{\omega,b}(x_i)] + C_U \sum_{j=1}^{u} I[f_{\omega,b}(z_j)]$$

(7)

3. Weighted Twin Support Vector Machine with Universum (U-WTSVM)

From the above description, we know that U-TSVM uses hinge loss function just as U-SVM, so this algorithm is less robust, in other words, sensitive to noises. This is the common drawback of any learning method using hinge loss function. In order to avoid this drawback, we consider weighted TSVM idea to improve the algorithm by using weights to modify error variables. The proposed algorithm is called weighted U-TSVM (U-WTSVM) by us.

3.1 Linear U-WTSVM

The formulations of Linear U-WTSVM are expressed as follows

$$\min_{w_1,b_1,\xi,\psi} \; \frac{1}{2}\|Aw_1 + e_1 b_1\|^2 + c_1 \rho_1^T \xi + c_u e_u^T \psi$$

(8)

$$s.t. -(Bw_1 + e_2 b_1) + \xi \geq e_2, \xi \geq 0$$

(9)

$$(Uw_1 + e_u b_1) + \psi \geq (-1 + \epsilon)e_u, \psi \geq 0$$

(10)

And

$$\min_{w_2,b_2,\eta,\varphi} \; \frac{1}{2}\|Aw_2 + e_2 b_2\|^2 + c_2 \rho_2^T \eta + c_u e_u^T \varphi$$

(11)

$$s.t. -(Aw_2 + e_1 b_2) + \eta \geq e_1, \eta \geq 0$$

(12)

ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 2, No.8 , March 2014
ISSN : 2322-5157
www.ACSIJ.org

$$-(Uw_2 + e_u b_2) + \varphi \geq (-1 + \epsilon)e_u, \varphi \geq 0 \quad (13)$$

Where $\psi = (\psi_1, \dots, \psi_u)^T$, $\varphi = (\varphi_1, \dots, \varphi_u)^T$ and $\epsilon, c_1, c_2, c_u \in [0, +\infty]$ are the prior parameters. The Lagrangian based on the optimization problem (8)-(10) is given by

$$L(w_1, b_1, \xi, \psi, \alpha, \beta, \mu, \gamma) = \frac{1}{2}\|Aw_1 + e_1 b_1\|^2 +$$
$$c_1 \rho_1^T \xi + c_u e_u^T \psi - \alpha^T(-(Bw_1 + e_2 b_1) + \xi - e_2) - \quad (14)$$
$$\beta^T \xi - \mu^T((Uw_1 + e_u b_1) + \psi + (1 - \epsilon)e_u) - \gamma^T \psi$$

where $\alpha, \beta, \mu, \gamma$ are the members of the vectors of Lagrange multipliers. The Karush-Kuhn-Tucker (KKT) conditions for (14) are obtained as follows

$$\frac{\partial L}{\partial w_1} = A^T(Aw_1 + e_1 b_1) + B^T \alpha - U^T \mu = 0 \quad (15)$$

$$\frac{\partial L}{\partial b_1} = e_1^T(Aw_1 + e_1 b_1) + e_2^T \alpha - e_u^T \mu = 0 \quad (16)$$

$$\frac{\partial L}{\partial \xi} = c_1 \rho_1 - \alpha - \beta = 0 \quad (17)$$

$$\frac{\partial L}{\partial \psi} = c_u e_u - \mu - \gamma = 0 \quad (18)$$

Since $\beta, \gamma \geq 0$, (17) and (18) turn to be

$$0 \leq \alpha \leq c_1 \rho_1 \quad \text{and} \quad 0 \leq \mu \leq c_u e_u$$

Combining (15) and (16) leads to

$$[A^T \ e_1^T]^T [A \ e_1][w_1 \ b_1]^T + [B^T \ e_2^T]^T \alpha$$
$$- [U^T \ e_u^T]^T \mu = 0 \quad (19)$$

Let $H = [A \ e_1]$, $G = [B \ e_2]$, $0 = [U \ e_u]$, equation (19) can be rewritten as

$$H^T H v_1 + G^T \alpha - O^T \mu = 0$$
$$\text{i.e., } v_1 = -(H^T H)^{-1}(G^T \alpha - O^T \mu). \quad (20)$$

We can use (8)–(10) to solve the following convex QPP, which is the Wolfe's dual of $\mathcal{U}$-WTSVM

$$\max_{\alpha, \mu} -\frac{1}{2}(\alpha^T G - \mu^T O)(H^T H)^{-1}(G^T \alpha - O^T \mu) + e_2^T \alpha$$
$$+ (\varepsilon - 1)e_u^T \mu$$

$$\text{s.t. } 0 \leq \alpha \leq c_1 \rho_1, \qquad 0 \leq \mu \leq c_u e_u, \quad (21)$$

Similarly, the dual of (11)–(13) is formulated as

$$\max_{\alpha, \mu} -\frac{1}{2}(\lambda^T P - \upsilon^T S)(Q^T Q)^{-1}(P^T \lambda - S^T \upsilon) + e_1^T \lambda$$
$$+ (\varepsilon - 1)e_u^T \upsilon$$

$$\text{s.t. } 0 \leq \lambda \leq c_2 \rho_2, \qquad 0 \leq \upsilon \leq c_u e_u, \quad (22)$$

where $Q = [A \ e_2]$, $P = [B \ e_1]$, $S = [U \ e_u]$, the augmented vector $[w_1 \ b_1]^T$ is given by

$$v_2 = -(Q^T Q)^{-1}(P^T \lambda - S^T \upsilon) \quad (23)$$

Once the augmented vectors of v_1 and v_2 are known, the two separating planes are obtained. The class of an unknown data point $x \epsilon R^n$ is determined as

$$class(x) = \underset{i=1,2}{\operatorname{argmin}}\{d_1(x), d_2(x)\} \quad (24)$$

Where $d_1(x) = |w_1^T x + b_1|$, $d_2(x) = |w_2^T x + b_2|$, where $|\cdot|$ denotes the perpendicular distance of the point x from the planes. Following the same idea, we extend $\mathcal{U}$-WTSVM to its nonlinear version.

3.2 Nonlinear Kernel Classifier

In order to extend our algorithm to nonlinear cases, we considering the following kernel-based surfaces instead of planes

$$K(x^T, C^T)k_1 + b_1 = 0, \quad K(x^T, C^T)k_2 + b_2 = 0 \quad (25)$$

Where $C^T = [A \ B]^T$ and K is a chosen kernel function. Then the optimization problems of nonlinear $\mathcal{U}$-WTSVM are constructed as follows

$$\min_{k_1, b_1, \xi, \psi} \frac{1}{2}\|K(A, C^T)k_1 + e_1 b_1\|^2 + c_1 \rho_1^T \xi \quad (26)$$
$$+ c_u e_u^T \psi$$

$$s.t. \quad -(K(B, C^T)k_1 + e_2 b_1) + \xi \geq e_2, \quad \xi \geq 0$$

$$(K(U, C^T)k_1 + e_2 b_1) + \psi \geq (-1 + \epsilon)e_u, \quad \psi \geq 0$$

And

$$\min_{k_2, b_2, \eta, \varphi} \frac{1}{2}\|K(B, C^T)k_2 + e_2 b_2\|^2 + c_2 \rho_2^T \eta \quad (27)$$
$$+ c_u e_u^T \varphi$$

$$s.t. \quad (K(A, C^T)k_2 + e_1 b_2) + \eta \geq e_1, \quad \eta \geq 0$$

$$-(K(U, C^T)k_2 + e_1 b_2) + \varphi \geq (-1 + \epsilon)e_u, \varphi \geq 0$$

The Wolfe dual of the problem (26) is formulated as follow

$$\max_{\alpha, \mu} -\frac{1}{2}(\alpha^T G_\phi - \mu^T O_\phi)(H_\phi^T H_\phi)^{-1} \quad (28)$$
$$(G_\phi^T \alpha - O_\phi^T \mu) + e_2^T + (\varepsilon - 1)e_u^T \mu$$

$$s.t. \, 0 \leq \alpha \leq c_1 \rho_1, \ 0 \leq \mu \leq c_u e_u,$$

where $H_\phi = [K(A, C^T)\ e_1]$, $G_\phi = [K(B, C^T)\ e_2]$, $O_\phi = [K(U, C^T)\ e_u]$, The augmented vector $v_1 = [k_1\ b_1]^T$ can be rewritten as

$$H_\phi^T H_\phi \rho_1 + G_\phi^T \alpha - O_\phi^T \mu = 0$$

i.e., $v_1 = (H_\phi^T H_\phi)^{-1}(G_\phi^T \alpha - O_\phi^T \mu)$ \hfill (29)

In a similar way, we can obtain the Wolfe's dual of the optimization problem (27)

$$\max_{\lambda, v} -\frac{1}{2}(\lambda^T P_\phi - v^T S_\phi)(Q_\phi^T Q_\phi)^{-1}$$

$$(P_\phi^T \lambda - S_\phi^T v) + e_1^T \lambda + (\varepsilon - 1)e_u^T v$$ \hfill (30)

$$s.t.\ 0 \le \lambda \le c_2 \rho_2,\ 0 \le v \le c_u e_u,$$

where $Q_\phi = [K(A, C^T)\ e_2]$, $P_\phi = [K(B, C^T)\ e_1]$, $S_\phi = [K(U, C^T)\ e_u]$, The augmented vector $v_2 = [k_2\ b_2]^T$ is given by

$$v_2 = -(Q_\phi^T Q_\phi)^{-1}(P_\phi^T \lambda - S_\phi^T v)$$ \hfill (31)

4. Experiments

To check the performances of the proposed U-WTSVM, we compare it with TSVM and U-TSVM on several datasets, including toy data and UCI datasets. In experiments, for each SVM algorithm for nonlinear kernels, a Gaussian kernel (i.e. $K(x, x_i) = exp(-\|x - x_1\|^2/2\sigma^2)$)was selected. We implemented all algorithms in MATLAB 2010 and carried out experiments on a PC with an Intel (R) Core 2 Duo processor (2.79 GHz), 4 GB of RAM. The parameters

c_1, c_2, c_u and the RBF kernel parameter σ are selected from the set of values $\{2^i | i = -7, ..., 7\}$ by tuning a set comprising of random 10 percent of the sample set. In the experiments we set $c_1 = c_2 = c_u$. Once the parameters are determined, the tuning set is returned to the sample set to learn the final decision function.
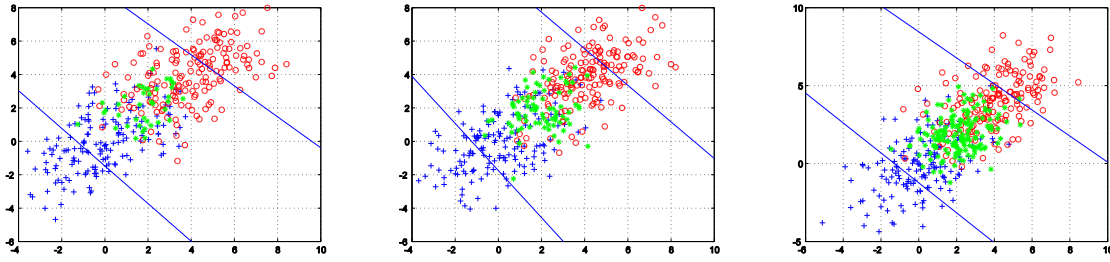
4.1 Toy data

To give an intuitive performance of U-WTSVM in different numbers of Universum data, we firstly use a toy 2D data to test the influence of Universum data to the accuracy. The positive data and negative data are generated randomly from two Gaussian distributions. Universum data is also generated by Gaussian distribution. We use 200 positive data and 200 negative data as the data set, 40, 80, 160 unlabeled data as Universum samples respectively. For the experiment, we use the 20% of the points for training and others for testing. We select linear kernel for the toy dataset. The comparative results of TSVM, U-TSVM and U-WTSVM are shown in Fig 1. The second toy data is a nonlinear separated example, Fig 2 shows the results of TSVM, U-TSVM and U-WTSVM.

From the Fig 1. It is easy to see that Universum data can indeed help our algorithm to seek more reasonable classifier. With the increase of the number of Universum data, the average accuracies of the three algorithms in the toy data are constantly being improved. Fig 2 shows that the knowledge embedded in the Universum points indeed helps the U-WTSVM to have better performance than its original model. From Fig 3, with the increase of Universum data, the average accuracies of the two algorithms are obviously improved, which means that Universum data can indeed help our algorithm to seek more reasonable classifier.
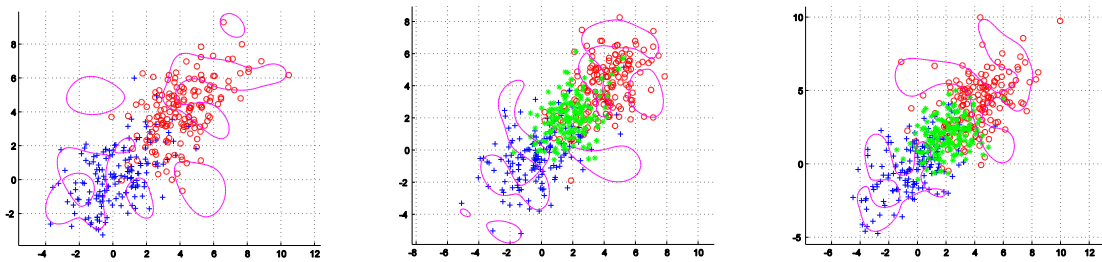
NU=40   NU=80   NU=160
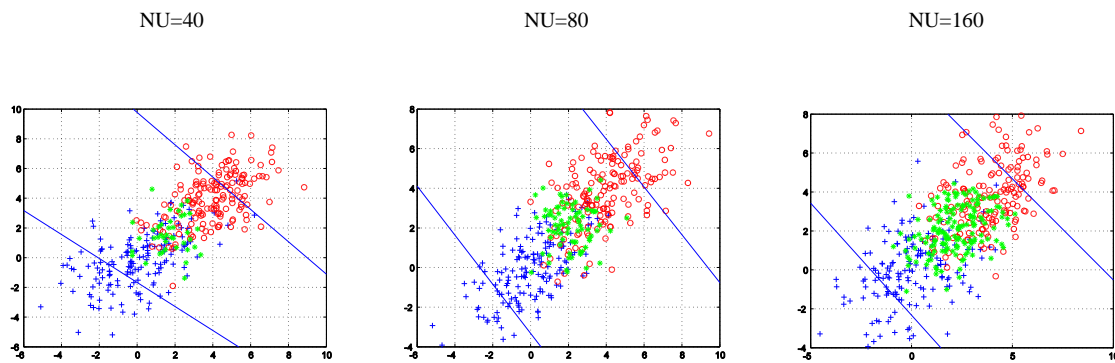
(a)The result of U-TSVM on the training set.

(b)The result of Ʉ-WTSVM on the training set.

Figure 1: The performance comparison of Ʉ-TSVM and Ʉ-WTSVM on toy data. positive points(blue 'o'), negative points(red '+'), Universum data(green '*') , two nonparallel hyperplanes(blue solid line).

(a)The result of TSVM on the training set.   (b)The result of Ʉ-TSVM on the training set.   (c)The result of Ʉ-WTSVM on the training set.
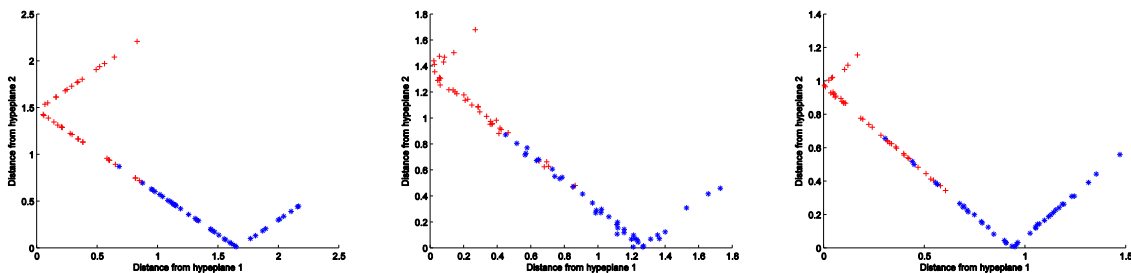
(d)Two-dimensional projections of TSVM.   (e) Two-dimensional projections of Ʉ-TSVM. (f) Two-dimensional projections of Ʉ-WTSVM.

Figure 2: The performance of TSVM, Ʉ-TSVM and Ʉ-WTSVM in the RBF case. positive points(blue 'o'), negative points(red '+'), Universum data(green '*') , pink solid curves are the hyperplanes of TSVM, Ʉ-TSVM and Ʉ-WTSVM, respectively.

4.2 UCI datasets

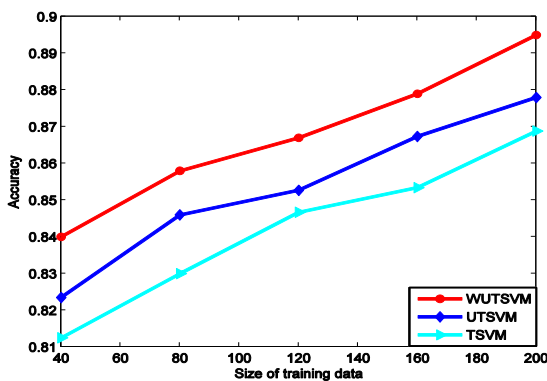In this section, we perform these methods on the UCI datasets. For each dataset, we randomly select the same number of data from different classes to compose a dataset. Fifty percent of each extracted dataset are for training, others for testing, and also 50% of training data are used to generate Universum data .There are many methods to collect Universum data in practice. Here, we use each Universum example is generated by selecting the data from two different categories of training datasets and then combined with a mean coefficient. The parameters' selection of models uses 10-fold cross validation method mentioned above. Each experiment repeats 10 times. The final results are shown in the Table 1. From Table 1, we

Figure 3: The accuracy rates of TSVM, Ʉ-TSVM and Ʉ-WTSVM in the toy data.

can find that the Ʊ-WTSVM which adds the Universum data outperforms the normal TSVM and Ʊ-TSVM in most cases. And then we use 10 times trials results of Ʊ-WTSVM, Ʊ-TSVM and TSVM with t-test. The experimental results H and P show in Table 2. Compare with Ʊ-TSVM and TSVM, it is easy to find that the probability of Ʊ-WTSVM is extremely low, this means that the accuracy of Ʊ-WTSVM are significantly different with the other two.

Table 1:The testing accuracy and training times on UCI datasets in the case of RBF.

Datasets	Ʊ-WTSVM accuracy time(s)	Ʊ-TSVM accuracy time(s)	TSVM accuracy time(s)
Liver (345×7)	80.00% 0.1537	76.52% 0.1944	67.83% —
Diabetes (768×9)	76.56% 1.0162	75% 1.0162	72.92% —
BreastCancer (569×31)	76.84% 0.8255	76.32% 0.9207	75.26% —
AustralianCredit (690×15)	68.26% 1.3435	67.39% 1.7382	66.52% —
SPECTHeart (569×31)	58.43% 0.2896	56.18% 0.3308	57.30% —
Statlog (690×15)	66.67% 1.1039	65.22% 1.6372	63.77% —
Pima (768×9)	82.81% 0.4604	81.51% 0.6470	81.25% —
Car (1728×7)	70.72% 2.4714	70.37% 2.6359	71.41% —
Ionosphere (351×34)	72.42% 0.6344	70.44% 0.6476	69.03% —

Table 2: The comparison results of Ʊ-WTSVM, Ʊ-TSVM and TSVM on t-test.

Datasets	Ʊ-WTSVM with Ʊ-TSVM	Ʊ-WTSVM with TSVM
Liver	(1, 2.52E-06)	(1, 3.05E-12)
Diabetes	(1, 0.001721)	(1, 1.53E-06)
BreastCancer	(0, 0.181566)	(1, 0.009097)
eAustralianCredit	(1, 0.020619)	(1, 0.004246)
SPECTHeart	(1, 0.000709)	(1, 0.017269)
Statlog	(0, 0.318653)	(1, 5.05E-05)
Pima	(0, 0.083544)	(0, 0.067579)
Car	(1, 0.049098)	(0, 0.159092)
Ionosphere	(1, 0.029548)	(1, 0.041772)

5. Conclusion

Universum, which is defined as the sample that does not belong to either class of the classification problem of interest, has been proved to be helpful in supervised learning. In this paper, we have proposed a new weighted twin support vector machine with Universum algorithm (Ʊ-WTSVM) which is able to improve the classification performance.

Experimental results revealed that Ʊ-WTSVM is better than Ʊ-TSVM in terms of both classification effectiveness and lower computational cost. However, although Ʊ-WTSVM performs faster than TSVM, the limitation is that it cannot handle large-scale problems. Thus, further work we will use the Ʊ-WTSVM to solve real large scale classification problems. And how to generate or select Universum data is also a main topic in the future.

References

[1] J. Weston, R. Collobert, F. Sinz, L. Bottou, V. Vapnik, "Inference with the Universum", ICML '06: Proceedings of the 23rd international conference on machine learning, ACM, 2006, pp. 1009–1016.

[2] F. H. Sinz, O. Chapelle, A. Agarwal, B. Scholkopf, "An analysis of inference with the universum", Advances in Neural Information Processing Systems 20, MIT Press, 2008, pp. 1369–1376.

[3] D. Zhang, J. Wang, F. Wang, C. Zhang, "Semi-supervised classification with universum", SIAM International Conference on Data Mining(SDM), 2008, pp. 323-333.

[4] S. Chen, C. Zhang, "Selecting informative universum sample for semisupervised learning". IJCAI, 2009, pp. 1016–1021.

[5] V. Cherkassky, S. Dhar, W. Dai, "Practical conditions for effectiveness of the universum learning", IEEE Transactions on Neural Networks , Vol. 22, No. 8 , 2011, pp. 1241–1255.

[6] C. Shen, P. Wang, F. Shen, H. Wang, "Uboost: Boosting with the universum", IEEE Transaction on Pattern Analysis and Machine Intelligence. Vol. 34, No. 4, 2012 , pp.825-832.

[7] Z.-Q. Qi, Y.-J. Tian, Y. Shi, "Twin support vector machine with Universum data", IEEE Transactions on Neural Networks , Vol. 36, 2012 , pp. 112–119.

[8] Jayadeva, R. Khemchandani, S. Chandra, "Twin support vector machines for pattern classification", IEEE Transactions on Pattern Analysis and Machine Intellegence, Vol. 29, No. 5, 2007, pp. 905–910.

[9] Y.-H. Shao, C.-H. Zhang, X.-B. Wang, N.-Y. Deng, "Improvements on twin support vector machines", IEEE Transactions on Neural Networks, Vol. 22, No. 6, 2011, pp. 962–968.

[10] M. A. Kumar, M. Gopal, "Application of smoothing technique on twin support vector machines", Pattern Recognition Letters, Vol. 29, No. 13, 2008, pp. 1842–1848.

[11] R. Khemchandani, Jayadeva, S. Chandra, "Optimal kernel selection in twin support vector machines", Optimization Letters, Vol. 3, No. 1, 2009, pp. 77– 88.

[12] N. Deng, Y. Tian, Support vector machines: Theory, Algorithms and Extensions, Science Press, China, 2009.

Shuxia Lu   Professor of the Faculty of Mathematics and Computer Science, Hebei University. Received the B.Sc. and M.Sc. degrees in Mathematics from Hebei University, Baoding, China, in 1988 and 1991, respectively, and the Ph.D. degree from Hebei University, Baoding, China, in 2007. Her main research interests include machine learning and computational intelligence, SVMs.

Le Tong Received the B.Sc. degree in Mathematics and its Application from Baoding Normal University, PR China, in 2012. She has been a M.Sc. degree candidate in applied mathematics from Hebei University, Baoding, China. Her research interests include Machine Learning and SVM.