IJAMML 5:1 (2016) 1-13 September 2016
ISSN: 2394-2258
Available at http://scientificadvances.co.in
DOI: http://dx.doi.org/10.18642/ijamml_7100121598

LEAST SQUARES UNIVERSUM TSVM

Yanmeng Li and Liya Fan

School of Mathematics Sciences, Liaocheng University, 252059, P. R. China

Abstract

Supervised learning problem with Universum data is a new research subject in machine learning. Universum data, which are not belonging to any class of the classification problem of interest, has been proved very helpful in learning. For data classification with Universum data, a novel quick classifier is proposed in this paper and named as least squares Universum twin support vector machine (LS-U-TSVM). The main advantage of the proposed method is that the running time is shorten greatly by using least squares technique. Experiment results indicate that the proposed LSU-TSVM is an effective and competitive classifier for data classification with Universum data.

Keywords: twin support vector machine, Universum data, least squares technology, classification accuracy.

1. Introduction

Supervised learning problem with Universum data is a new research subject in machine learning. The concept of Universum data was firstly introduced by Weston et al. [1] in 2006, which is defined as the data not

*Corresponding author.
E-mail address: fanliya63@126.com (Liya Fan).

2010 Mathematics Subject Classification: 68W40, 68Q25.
Submitted by Mehmet Koc.
Received November 22, 2015

belonging to any of the classes the learning task concerns. For instance, considering the classification of 5 against 8 in handwritten digits recognition, "0, 1, 2, 3, 4, 6, 7, 9" can be considered as Universum data. Since they are not required to have the same distribution with the training data, the Universum data are able to show some prior information for the possible classifiers. Several works have been done by using the Universum data in machine learning, such as support vector machine with Universum data (U-SVM) [1], least squares U-SVM (LS-U-SVM) [2], and twin support vector machine with Universum data (U-TSVM) [3]. Other literatures also can be found in [4, 5].

It is known that in the past decades, numerous SVM-type supervised classification methods have been proposed and successfully applied to many fields. All these methods can be divided into two types. One is solving the dual problems, such as twin support vector machine (TSVM) [6] and improved TSVM (ITSVM) [7]. Another is solving the primal problems, such as least square SVM (LS-SVM) [8], sparse least square SVM [9], least squares twin multi-class SVM [10], structural least square TSVM [11] and so on [12-14].

Motivated by the above works, in this paper, we will study the least square version of U-TSVM and propose a new classification method named as least square Universum twin support vector machine (LS-U-TSVM). The proposed method is also an extension and improvement of LS-U-SVM. The use of least squares technique aims to reduce the running time (the sum of training time and testing time) of classifiers by avoiding quadratic programming problems (QPPs). We only study the linear version of LS-U-TSVM in this paper. With the kernel skill, we can also research the nonlinear version. In order to verify the effectiveness of LS-U-TSVM, a series of comparative experiments with linear TSVM and linear U-TSVM are performed on six experiment datasets taken from UCI database [15].

The rest of the paper is organized as follows. In Section 2, the background and related works are introduced. In Section 3, LS-U-TSVM is proposed with detailed derivation. Experiments and results analysis are performed in Section 4 and some conclusions are given in Section 5.

2. Background and Related Works

In this section, we recall briefly twin support vector machine (TSVM) and twin support vector machine with Universum data (U-TSVM), for details, see [3, 6]. Let $T = \{(x_i, y_i)\}_{i=1}^{l}$ be a set of binary data, where $x_i \in R^n$ and $y_i \in \{\pm 1\}$ are the input sample and class label of the i-th data, respectively. Let l_1 and l_2 be the numbers of positive and negative samples, respectively, and $l = l_1 + l_2$. Let $\{x_i\}_{i=l+1}^{l+u} \in R^n$ be a set of Universum data, which are not belonging to both positive and negative classes. We denote by $A \in R^{l_1 \times n}$, $B \in R^{l_2 \times n}$, and $U \in R^{u \times n}$ the matrices composed of positive, negative, and Universum data, respectively. Linear TSVM and linear U-TSVM complete the classification of data by seeking a pair of nonparallel hyperplanes

$$f_+(x) = w_+^T x + b_+ = 0, \; f_-(x) = w_-^T x + b_- = 0, \tag{1}$$

where $w_+, w_- \in R^n$ are normal vectors and $b_+, b_- \in R$ are thresholds. But their starting points are different.

2.1. TSVM

The basic idea of linear TSVM is constructing a pair of nonparallel hyperplanes defined by (1) such that each one is as close as possible to one class, and as far as possible from the other class. A new input sample will be assigned to one of the classes depending on its proximity to each hyperplane. For this end, linear TSVM considers the following two quadratic programming problems (QPPs):

$$\min_{w_+, b_+, \xi} \frac{1}{2}\|Aw_+ + e_+ b_+\|^2 + c_1 e_-^T \xi$$

$$s.t. -(Bw_+ + e_- b_+) + \xi \geq e_-, \quad \xi \geq 0, \tag{2}$$

$$\min_{w_+, b_+, \xi} \frac{1}{2}\|Bw_- + e_- b_-\|^2 + c_2 e_+^T \eta$$

$$s.t. -(Aw_- + e_+ b_-) + \eta \geq e_+, \quad \eta \geq 0, \tag{3}$$

where $c_1, c_2 > 0$ are trade-off parameters, $\xi \in R^{l_2}$, $\eta \in R^{l_1}$ are slack vectors and $e_+ \in R^{l_1}$, $e_- \in R^{l_2}$ are vectors of ones. By solving, respectively, the Wolfe dual forms of the problems (2) and (3):

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

$$s.t.\ 0 \leq \alpha \leq c_1 e_-,$$

$$\max_{\beta} e_+^T \beta - \frac{1}{2}\beta^T H(G^T G)^{-1} H^T \beta$$

$$s.t.\ 0 \leq \beta \leq c_2 e_+,$$

we can obtain (w_+, b_+) and (w_-, b_-), where $G = [B\ e_-] \in R^{l_2 \times (n+1)}$, $H = [A\ e_+] \in R^{l_1 \times (n+1)}$ and $\alpha \in R^{l_2}$, $\beta \in R^{l_1}$ are Lagrange multiplier vectors. A new input sample $\tilde{x} \in R^n$ can be assigned the class k depending on which of the two hyperplanes is closer to, that is, $k = \arg\min_{+,-}\{\frac{|f_+(\tilde{x})|}{\|w_+\|}, \frac{|f_-(\tilde{x})|}{\|w_-\|}\}$.

2.2. U-TSVM

Similar to linear TSVM, linear U-TSVM is also constructing a pair of nonparallel hyperplanes defined by (1) such that each one is as close as possible to one class, and as far as possible from the other class. But in linear U-TSVM, in order to improve the classification performance of classifiers, the Universum data are used for training classifiers. For this end, linear U-TSVM considers the following two QPPs:

$$\min_{w_+, b_+, \xi, \psi} \frac{1}{2}\|Aw_+ + e_+b_+\|^2 + c_1 e_-^T \xi + c_u e_u^T \psi$$

$$s.t. \quad -(Bw_+ + e_-b_+) + \xi \geq e_-,$$

$$(Uw_+ + e_u b_+) + \psi \geq (-1+\varepsilon)e_u,$$

$$\xi \geq 0,\ \psi \geq 0, \tag{4}$$

$$\min_{w_-, b_-, \eta, \psi^*} \frac{1}{2}\|Bw_- + e_-b_-\|^2 + c_2 e_+^T \eta + c_u e_u^T \psi^*$$

$$s.t. \quad (Aw_- + e_+b_-) + \eta \geq e_+,$$

$$-(Uw_- + e_u b_-) + \psi^* \geq (-1+\varepsilon)e_u,$$

$$\eta \geq 0,\ \psi^* \geq 0, \tag{5}$$

where $c_1, c_2, c_u, \varepsilon > 0$ are trade-off parameters, $\xi \in R^{l_2}$, $\eta \in R^{l_1}$, $\psi, \psi^* \in R^u$ are slack vectors and $e_u \in R^u$ is the vector of ones. By solving, respectively, the Wolfe dual forms of the problems (4) and (5):

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

$$s.t.\ 0 \leq \alpha \leq c_1 e_-,$$

$$0 \leq \mu \leq c_\mu e_u, \tag{6}$$

$$\max_{\lambda, \upsilon} -\frac{1}{2}(\lambda^T H - \upsilon^T O)(G^T G)^{-1}(H^T \lambda - O^T \nu) + e_+^T \lambda + (\varepsilon - 1)e_u^T \nu$$

$$s.t.\ 0 \leq \lambda \leq c_1 e_+,$$

$$0 \leq \nu \leq c_\mu e_u, \tag{7}$$

we can get (w_+, b_+) and (w_-, b_-), where $O = [U\ e_u] \in R^{u \times (n+1)}$ and $\alpha \in R^{l_2}$, $\lambda \in R^{l_1}$, $\mu, \nu \in R^u$ are Lagrange multiplier vectors. A new input $\tilde{x} \in R^u$ can be assigned the class k depending on which of the two hyperplanes is closer to, that is, $k = \arg\min_{+,-}\{\frac{|f_+(\tilde{x})|}{\|w_+\|}, \frac{|f_-(\tilde{x})|}{\|w_-\|}\}$.

3. Least Squares Universum TSVM (LS-U-TSVM)

Because the time of solving a QPP is about $O(n^3)$, where n is the number of training samples, we know from the dual problems (6) and (7) that the training time of U-TSVM is about $O((l_1+u)^3)+O((l_2+u)^3)$. If the number u of Universum data is more, the training time will be very long. In addition, in the process of solving the problems (6) and (7), the singularity problem of the matrices H^TH and G^TG may be existed. In order to avoid QPPs and reduce the running time of U-TSVM, in this section, we consider the least squares version of U-TSVM and propose a novel quick classification method named as least squares Universum twin support vector machine (LS-U-TSVM). For this end, we modify the problems (6) and (7) into the following forms:

$$\begin{aligned}\min_{w_+,b_+,\xi,\psi}\ & \frac{1}{2}\|Aw_+ + e_+b_+\|^2 + \frac{c_1}{2}\xi^T\xi + \frac{c_u}{2}\psi^T\psi \\ s.t.\quad & -(Bw_+ + e_-b_+) + \xi = e_-, \\ & (Uw_+ + e_ub_+) + \psi = (-1+\varepsilon)e_u, \end{aligned} \tag{8}$$

$$\begin{aligned}\min_{w_-,b_-,\eta,\psi^*}\ & \frac{1}{2}\|Bw_- + e_-b_-\|^2 + \frac{c_2}{2}\eta^T\eta + \frac{c_u}{2}{\psi^*}^T\psi^* \\ s.t.\quad & (Aw_- + e_+b_-) + \eta = e_+, \\ & -(Uw_- + e_ub_-) + \psi^* = (-1+\varepsilon)e_u, \end{aligned} \tag{9}$$

where the equality constraints instead of inequality constraints and the square of 2-norm of slack variables with weight $\frac{c_1}{2}$, $\frac{c_2}{2}$, and $\frac{c_u}{2}$ instead of 1-norm with weight c_1, c_2, and c_u. It is evident that the problems (8) and (9) can be transformed into the following unconstrained optimization problems:

$$\min_{v_+} F_1(v_+) = \frac{1}{2}\|Hv_+\|^2 + \frac{c_1}{2}\|Gv_+ + e_-\| + \frac{c_u}{2}\|Ov_+ - (-1+\varepsilon)e_u\|, \tag{10}$$

$$\min_{v_-} F_2(v_-) = \frac{1}{2}\|Gv_-\|^2 + \frac{c_2}{2}\|Hv_- - e_+\| + \frac{c_u}{2}\|Ov_- + (-1+\varepsilon)e_u\|, \quad (11)$$

where $v_+ = [w_+^T, b_+]^T$ and $v_- = [w_-^T, b_-]^T$. Letting $\frac{dF_1(v_+)}{dv_+} = \frac{dF_2(v_-)}{dv_-} = 0$, we can deduce that

$$Mv_+ + c_1G^Te_- + c_u(1-\varepsilon)O^Te_u = 0,$$

$$Mv_- - c_2H^Te_+ - c_u(1-\varepsilon)O^Te_u = 0, \quad (12)$$

where $M = H^TH + c_1G^TG + c_uO^TO$. Without loss of generality, we assume the symmetric nonnegative definite matrix M is nonsingular; otherwise, it can be regularized, that is, it can be replaced by $M + \delta I$, where I is an identity matrix of appropriate dimensions and $\delta > 0$ is a sufficiently small number. Consequently, it can be deduced from (12) that

$$v_+^* = -M^{-1}[c_1G^Te_- + c_u(1-\varepsilon)O^Te_u],$$

$$v_-^* = M^{-1}[c_2H^Te_+ + c_u(1-\varepsilon)O^Te_u]. \quad (13)$$

After obtaining v_+^* and v_-^*, the class label $y_{\tilde{x}}$ of a new input sample $\tilde{x} \in R^n$ can by determined. Specific algorithm is as follows:

Algorithm 1. (LS-U-TSVM)

Step 1. Given a set of data $T = \{(x_i, y_i)\}_{i=1}^{l} \cup \{x_i\}_{i=l+1}^{l+u}$, where $x_i \in R^n$, $i = 1, \cdots, l+u$, $y_i \in \{\pm 1\}$, $i = 1, \cdots, l$ and $\{x_i\}_{i=l+1}^{l+u}$ are Universum data.

Step 2. Select suitable modelling parameters $c_1, c_2, c_u > 0$.

Step 3. Calculate the matrix M.

Step 4. Calculate v_+^* and v_-^* by (13).

Step 5. For a new input sample $\widetilde{x} \in R^n$, its class label $y_{\widetilde{x}}$ can be determined by

$$y_{\widetilde{x}} = \arg \min_{i=+,-} \{|w_+^{*T}\widetilde{x} + b_+^*)|, |w_-^{*T}\widetilde{x} + b_-^*)|\}.$$

4. Experiments

In order to verify the effectiveness of linear LS-U-TSVM, in this section, we perform a series of comparative experiments with linear U-TSVM and linear TSVM on classification accuracy (acc) and training time (time) by means of 6 experiment datasets taken from UCI database. All the experiments are implemented in Matlab (7.11.0) R2010b environment on a PC with an Intel P4 processor (2.30GHz) with 4GB RAM and the five-fold cross-validation method is used. The classification accuracy is defined by

$$acc = \frac{TP + TN}{TP + FP + TN + FN},$$

where $TP, TN, FP,$ and FN denote the numbers of true positive, true negative, false positive, and false negative, respectively.

For Wine and Iris datasets with 3 classes, we choose the first two classes as labelled data and the third class as Universum data. For Vchicle dataset with 4 classes, we consist of four experiment datasets. The first one selects the first two classes as labelled data and the third class as Universum data. The second one selects the first two classes as labelled data and the fourth class as Universum data. The third one chooses the first and the third classes as labelled data and the fourth class as Universum data. The last one chooses the second and the third classes as labelled data and the fourth class as Universum data. We select randomly 50 data for each class of Wine and Iris datasets and 150 data for each class of Vchicle dataset.

It is known that the performance of classifiers seriously depends on the choice of parameters. For comparing the optimal results, all the parameters involved in classifiers are selected optimally from $\{2^{-8}, \cdots, 2^{8}\}$ by grid search. For degrading the computational complexity of parameter selection, we take $c_1 = c_2$. The selected results are listed in Table 1, where Vchicle (a, b, c) denotes that the experiment dataset is composed of a, b, c three classes of Vchicle and the third class c as Universum data. The experiment results are listed in Table 2.

We can see from Table 2 that the classification accuracy of LS-U-TSVM is higher than that of U-TSVM except for Iris dataset and significantly higher than that of TSVM. For running time, LS-UTSVM is far more than U-TSVM and TSVM. Specifically, LS-U-TSVM is faster than TSVM about 32 times and than U-TSVM about 546 times the maximum and 146 times the minimum.

According to the above analysis, we can conclude that LS-U-TSVM is an effectively and competitively quick method for data classification with Universum data.

Table 1. Selected parameters for linear classifiers

Dataset	**Classifier**	**Parameters**	**Parameters values**
Wine	TSVM	c_1, c_2	2, 2
(150*14)	U-TSVM	c_1, c_2, c_u, ϵ	2, 2, 167, 0.2
	LS-U-TSVM	c_1, c_2, c_u, ϵ	2, 2, 167, 0.2
Iris	TSVM	c_1, c_2	1, 1
(150*5)	U-TSVM	c_1, c_2, c_u, ϵ	2, 2, 13, 0.2
	LS-U-TSVM	c_1, c_2, c_u, ϵ	30, 30, 80, 0.2
Vchicle (1,2,3)	TSVM	c_1, c_2	2, 2
(450*19)	U-TSVM	c_1, c_2, c_u, ϵ	58, 58, 61, 0.1
	LS-U-TSVM	c_1, c_2, c_u, ϵ	58, 58, 75, 0.1
Vchicle (1,2,4)	TSVM	c_1, c_2	58, 58
(450*19)	U-TSVM	c_1, c_2, c_u, ϵ	195, 195, 101, 0.1
	LS-U-TSVM	c_1, c_2, c_u, ϵ	122, 122, 132, 0.1
Vchicle (1,3,4)	TSVM	c_1, c_2	122, 122
(450*19)	U-TSVM	c_1, c_2, c_u, ϵ	195, 195, 10, 0.1
	LS-U-TSVM	c_1, c_2, c_u, ϵ	1, 1, 50, 0.2
Vchicle (2,3,4)	TSVM	c_1, c_2	122, 122
(450*19)	U-TSVM	c_1, c_2, c_u, ϵ	195, 195, 132, 0.1
	LS-U-TSVM	c_1, c_2, c_u, ϵ	1, 1, 70, 0.2

Table 2. Comparative results of three classifiers

Dataset	Classifier	Acc	Time(s)
Wine	TSVM	50.00	0.34
(150*14)	U-TSVM	58.75	9.36
	LS-U-TSVM	**65.00**	0.064
Iris	TSVM	50.00	0.35
(150*5)	U-TSVM	**60.00**	9.96
	LS-U-TSVM	50.00	0.065
Vchicle (1,2,3)	TSVM	50.15	2.15
(450*19)	U-TSVM	50.25	97.85
	LS-U-TSVM	**51.25**	0.18
Vchicle (1,2,4)	TSVM	50.33	2.01
(450*19)	U-TSVM	53.67	48.41
	LS-U-TSVM	**53.67**	0.14
Vchicle (1,3,4)	TSVM	50.00	1.05
(450*19)	U-TSVM	67.67	48.41
	LS-U-TSVM	**76.33**	0.16
Vchicle (2,3,4)	TSVM	50.00	2.18
(450*19)	U-TSVM	70.33	47.64
	LS-U-TSVM	**84.00**	0.17

5. Conclusion

In this paper, we propose an effective and quick classifier LS-U-TSVM for vector data classification with Universum data. The proposed method has two advantages. One is that the running time is shorten greatly by means of least squares technique. Another is that the classification performance can be improved by using the Universum data that are not belonging to any class. Experiment results show that the proposed LS-U-TSVM is an effectively and competitively quick

classification method. The next step of our work is to improve our model to obtain more effective method to solve the classification problems with Universum data and extend it to multi-class classification.

References

[1] J. Weston, R. Collobert, F. H. Sinz, L. Bottou and V. Vapnik, Inference with the Universum, In Proceedings of the 23rd International Conference on Machine Learning, (2006), 1009-1016.

[2] F. H. Sinz, O. Chapelle, A. Agarwal and B. Schlkopf, An analysis of inference with the Universum, In Advances in Neural Information Processing Systems 20 (2008), 1369-1376.

[3] Zhiquan Qi and Yingjie Tian, Twin support vector machine with Universum data, Neural Networks 36 (2012), 112-119.

[4] C. Shen, P. Wang, F. Shen and H. Wang, Boosting with the Universum, IEEE Transaction on Pattern Analysis and Machine Intelligence (2011).

[5] V. Cherkassky, S. Dhar and W. Dai, Practical conditions for effectiveness of the Universum learning, IEEE Transactions on Neural Networks 22(8) (2011), 1241-1255.

[6] Dr. Jayadeva, R. Khemchandani and S. Chandra, Twin support vector machines for pattern classification, IEEE Transactions on Pattern Analysis and Machine Intelligence 29(5) (2007), 905-910.

[7] Y. H. Shao, C. H. Zhang, X. B. Wang and N. Y. Deng, Improvements on twin support vector machines, IEEE Transactions on Neural Networks 22(6) (2011), 962-968.

[8] M. Arun Kumar and M. Gopal, Least squares twin support vector machines for pattern classification, Expert Systems with Applications 36 (2009), 7535-7543.

[9] Danilo Avilar Silva and Juliana Peixoto Silva, Novel approaches using evolutionary computation for sparse least square support vector machines, Neurocomputing 168(30) (2015), 908-916.

[10] A. Jalal Nasiri, Nasrollah Moghadam Charkari and Saeed Jalili, Least squares twin multi-class classification support vector machine, Pattern Recognition 48(3) (2015), 984-992.

[11] Yitian Xu, Xianli Pan, Zhijian Zhou, Zhiji Yang and Yuqun Zhang, Structural least square twin support vector machine for classification, Applied Intelligence 42(3) (2015), 527-536.

[12] Xiaopeng Hua and Shifei Ding, Weighted least squares projection twin support vector machines with local information, Neurocomputing 160(21) (2015), 228-237.

[13] Shifei Ding and Xiaopeng Hua, Recursive least squares projection twin support vector machines for nonlinear classification, Neurocomputing 130(23) (2014), 3-9.

[14] Jianhui Guo, Ping Yi, Ruili Wang, Qiaolin Ye and Chunxia Zhao, Feature selection for least squares projection twin support vector machine, Neurocomputing 144(20) (2014), 174-183.

[15] http://www.ics.uci.edu/mlearn/MLRepository.html, 1998.

■