International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.9, (2016), pp.379-388
http://dx.doi.org/10.14257/ijsip.2016.9.9.35

Linear One-Class Support Tensor Machine

Yanyan Chen[1,2] and Ping Zhong[1,*]

[1]*College of science, China Agricultural University,Beijing,100083,China*
[2]*College of Applied Science and Technology,Beijing Union University,Beijing 102200,China*
*yykjtyanyan@buu.edu.cn,*zping@cau.edu.cn*

Abstract

One-class support vector machine is an important and efficient classifier which is used in the situation that only one class of data is available, and the other is too expensive or difficult to collect. It uses vector as input data, and trains a linear or nonlinear decision function in vector space. However, there is reason to consider data as tensor. Tensor representation can make use of the structural information present in the data, which cannot be handled by the traditional vector based classifier. The significant benefit of using tensor as input is the reduction of the number of decision parameters, which can avoid the overfitting problems and especially suitable for small sample and large dimension cases. In this paper we have proposed a tensor based one-class classification algorithm named linear one-class support tensor machine. It aims to find a hyperplane in tensor space with maximal margin from the origin that contains almost all the data of the target class. We demonstrate the performance of the new tensor based classifier on several publicly available datasets in comparison with the standard linear one-class support vector machine. The experimental results indicate the validity and advantage of our tensor based classifier.

Keywords: *Support Vector Machine; One-Class Support Vector Machine; Support Tensor Machine; Linear One-Class Support Tensor Machine*

1. Introduction

The theory of Support Tensor Machine (STM) was proposed by Cai [1] and Tao [2-3]. Differ from popular existing learning algorithms taking vectors as input data, STM uses tensors. Similar to the framework of Support Vector Machine (SVM), STM aims at finding a maximum margin classifier in the tensor space. The advantage of tensor representation can be shortly summarized in the following two aspects. Firstly, tensor representation can greatly reduce the number of parameters estimated by SVM and especially be suitable for small sample and high dimension cases. Secondly, there is reason to consider data as tensors in real world. For example, grayscale image can be considered as second order tensor and color image can be represented as third order tensor. Tensor representation can make use of the structural information present in the data, while traditional vector algorithm cannot keep it efficiently. All these reasons lead us to concentrate on tensor based learning algorithms.

Recently, there have been a lot of interests in tensor based approaches. Cai *et. al.,* [4] applied STM for text classification; Tao *et. al.,* extended the classical linear C-SVM, v-SVM and least squares support vector machine to the general tensor form [3]; Zhang *et. al.,* generalized the vector-based learning algorithm twin support vector machine to the tensor-based method [5]; Hao *et. al.,* provided a novel linear support higher-order tensor machine for classification [6]; Reshma *et. al.,* developed a least squares variant of STM,

---

*Corresponding Author

ISSN: 2005-4254 IJSIP

termed as proximal support tensor machine [7], where the classifier is obtained by solving a system of linear equations rather than a quadratic programming problem, and applied to face recognition and handwriting recognition with good performance.

One-class support vector machine [8-9] is an important and widely used classifier. It can be used in which the negative samples are hardly collected or labeled, such as intrusion detection, fault detection and diagnosis, and the classification of remote sensing data [10-11]. In recent years, there have been more and more theory research and applications on one-class SVM [12-13]. However, using tensor as input data on the one-class classification has not been performed yet. In this paper, we focus on tensor based one-class classification problem, and propose a new one-class classifier termed as One-Class Support Tensor Machine.

The structure of the paper is as follows. A brief summary of some relevant concepts in STM and the standard one-class SVM is presented in Section 2. In Section 3, we formulate the tensor representation classification algorithm for one-class problems, which we call linear one-class STM, since we focus on linear classification problems. The experiments of the linear one-class STM on public datasets are described and compared with standard one-class SVM in Section 4. Finally, we draw conclusions of our work in Section 5.

2. Reviews of Relevant Research

Before presenting our work, we first briefly review the STM algorithm and the standard one-class SVM algorithm.

2.1. Support Tensor Machine

STM is similar with support vector machine, which was developed by Cai [1] and Tao [2-3] as a tensor generalization of SVM in the tensor space.

Suppose the training sample set $\{\mathbf{X}_i, y_i\}, i = 1\ldots l$, where $\mathbf{X}_i \in \mathbb{R}^{n_1} \otimes \mathbb{R}^{n_2}$ is the data point in order-2 tensor space, $y_i \in \{-1,1\}$ is the class associated with $\mathbf{X}_i$, $\mathbb{R}^{n_1}$ and $\mathbb{R}^{n_2}$ are two vector spaces. STM tries to find the following linear classifier in the tensor space

$$f(\mathbf{X}) = sgn\left(\mathbf{u}^T \mathbf{X} \mathbf{v} + b\right), \quad \mathbf{u} \in \mathbb{R}^{n_1}, \mathbf{v} \in \mathbb{R}^{n_2} \tag{1}$$

so that the two classes can be separated with maximum margin.

The objective function of linear STM can be stated as:

$$\begin{aligned} &\min_{\mathbf{u}\in\mathbb{R}^{n_1},\mathbf{v}\in\mathbb{R}^{n_2},\xi\in\mathbb{R}^{l},b\in\mathbb{R}} \frac{1}{2}\left\|\mathbf{u}\mathbf{v}^T\right\|^2 + C\sum_{i=1}^{l}\xi_i \\ &\qquad s.t. \quad y_i\left(\mathbf{u}^T\mathbf{X}_i\mathbf{v} + b\right) \geq 1 - \xi_i \\ &\qquad\qquad \xi_i \geq 0, i = 1,\ldots,l \end{aligned} \tag{2}$$

In order to solve the optimization problem (2), Cai [1] described a simple yet effective computational method. To fix $\mathbf{u}$ at first, for instance, let $\mathbf{u}$ be $(1,\ldots,1)^T$, the optimization problem (2) is identical to the standard SVM optimization problem with variable $\mathbf{v}$. And $\mathbf{v}$ can be solved by the same computational methods of SVM. Then $\mathbf{u}$ and $\mathbf{v}$ can be obtained by iteratively solving the standard SVM optimization problems. The convergence proof of the iterative computational method in linear STM was also provided by Cai [1].

2.2. One-Class Support Vector Machine

One-class support vector machine is a useful technique for data classification. It aims to learn a single class by determining a decision function with maximal margin from the origin that contains almost all the data of this class. We only review the linear standard one-class SVM in this section. Consider training data $\mathbf{x}_i \in \mathbb{R}^n, i = 1,...,l$ the decision hyperplane relative to the membership of the observation $\mathbf{x}$ to the considered class is given by:

$$f(\mathbf{x}) = sgn((\mathbf{w} \cdot \mathbf{x}) - \rho) \tag{3}$$

where parameters $\mathbf{w}$ and ρ result from the optimization problem:

$$\begin{aligned} \min_{\mathbf{w},\xi,\rho} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{\nu l}\sum_{i=1}^{l}\xi_i - \rho \\ s.t. \quad & (\mathbf{w} \cdot \mathbf{x}_i) \geq \rho - \xi_i \\ & \xi_i \geq 0, i = 1,...,l \end{aligned} \tag{4}$$

In practice, the quadratic program (4) is solved via its dual:

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j(\mathbf{x}_i \cdot \mathbf{x}_j) \\ s.t. \quad & 0 \leq \alpha_i \leq \frac{1}{\nu l} \\ & \sum_{i=1}^{l}\alpha_i = 1, i = 1,...,l \end{aligned} \tag{5}$$

The optimal normal vector is given by $\mathbf{w} = \sum_{i=1}^{l}\alpha_i\mathbf{x}_i$, where α_i is the solution of the dual problem (5) and training samples $\mathbf{x}_i$ with non-zero α_i are support vectors.

3. One-Class Support Tensor Machine

Our one-class Support Tensor Machine is fundamentally based on the same idea of the standard one-class SVM. In this section, we propose a linear one-class classifier based on tensor representation, which aims to find a hyperplane in tensor space with maximal margin from the origin that contains almost all the data of the target class.

Suppose we are given a set of training samples $\{\mathbf{X}_i\}, i = 1\ldots l$, each of the training sample $\mathbf{X}_i \in \mathbb{R}^{n_1} \otimes \mathbb{R}^{n_2}$ is the data point in order-2 tensor space, where $\mathbb{R}^{n_1}$ and $\mathbb{R}^{n_2}$ are two vector spaces.

As we discussed before, a linear one-class classifier in the tensor space can be naturally represented through matrix inner product as follows:

$$f(\mathbf{X}) = sgn((\mathbf{u}\mathbf{v}^T \cdot \mathbf{X}) - \rho), \quad \mathbf{u} \in \mathbb{R}^{n_1}, \mathbf{v} \in \mathbb{R}^{n_2} ? \tag{6}$$

Thus, the linear one-class STM can be given by solving the following optimal quadratic programming problem:

$$\begin{aligned} \min_{\mathbf{u}\in\mathbb{R}^{n_1},\mathbf{v}\in\mathbb{R}^{n_2},\xi\in\mathbb{R}^l,\rho\in\mathbb{R}} \quad & \frac{1}{2}\|\mathbf{u}\mathbf{v}^T\|^2 + \frac{1}{\nu l}\sum_{i=1}^{l}\xi_i - \rho \\ s.t. \quad & (\mathbf{u}\mathbf{v}^T \cdot \mathbf{X}_i) \geq \rho - \xi_i \\ & \xi_i \geq 0, i = 1,\ldots,l \end{aligned} \tag{7}$$

The Lagrangian corresponding to the optimization problem (7) is given by

$$L(\mathbf{u},\mathbf{v},\xi,\rho,\alpha,\beta)=\frac{1}{2}\left\|\mathbf{u}\mathbf{v}^T\right\|^2+\frac{1}{\nu l}\sum_{i=1}^{l}\xi_i-\rho-\sum_{i=1}^{l}\alpha_i\left(\left(\mathbf{u}\mathbf{v}^T\cdot\mathbf{X}_i\right)-\rho+\xi_i\right)-\sum_{i=1}^{l}\beta_i\xi_i \tag{8}$$

where $\alpha_i,\beta_i\geq 0, i=1,...,l$ are Lagrange multipliers, for each of the inequality constrains.

Note that

$$\frac{1}{2}\left\|\mathbf{u}\mathbf{v}^T\right\|^2=\frac{1}{2}\left(\mathbf{v}^T\mathbf{v}\right)\left(\mathbf{u}^T\mathbf{u}\right) \tag{9}$$

The Lagrangian (8) can be rewritten as

$$L(\mathbf{u},\mathbf{v},\xi,\rho,\alpha,\beta)=\frac{1}{2}\left(\mathbf{v}^T\mathbf{v}\right)\left(\mathbf{u}^T\mathbf{u}\right)+\frac{1}{\nu l}\sum_{i=1}^{l}\xi_i-\rho$$

$$-\sum_{i=1}^{l}\alpha_i\left(\mathbf{u}^T\mathbf{X}_i\mathbf{v}-\rho+\xi_i\right)-\sum_{i=1}^{l}\beta_i\xi_i \tag{10}$$

According to the KKT necessary and sufficient optimality conditions, we differentiate the primal variables $\mathbf{u},\mathbf{v},\xi_i,\rho$, and equate them to zero, then we have

$$\mathbf{u}=\frac{1}{\left\|\mathbf{v}\right\|^2}\sum_{i=1}^{l}\alpha_i\mathbf{X}_i\mathbf{v} \tag{11}$$

$$\mathbf{v}=\frac{1}{\left\|\mathbf{u}\right\|^2}\sum_{i=1}^{l}\alpha_i{\mathbf{X}_i}^T\mathbf{u} \tag{12}$$

$$\sum_{i=1}^{l}\alpha_i=1, \alpha_i=\frac{1}{\nu l}-\beta_i\leq\frac{1}{\nu l} \tag{13}$$

From Equations (11) and (12), we see that $\mathbf{u}$ and $\mathbf{v}$ are dependent on each other, and they cannot be solved independently. Hence, we resort to the alternating projection method for solving this optimization problem. The method can be described as follows.

First we fix $\mathbf{u}$. Let $\mu_1=\left\|\mathbf{u}\right\|^2$ and $\mathbf{x}_i=\mathbf{X}_i^T\mathbf{u}$, according to (7), we can construct the optimal quadratic programming problem to solve $\mathbf{v}$ and $\left\|\mathbf{v}\right\|^2$:

$$\begin{aligned}\min_{\mathbf{v}\in\mathbb{R}^{n_2},\xi\in\mathbb{R}^l,\rho\in\mathbb{R}}\ & \frac{1}{2}\mu_1\left\|\mathbf{v}\right\|^2+\frac{1}{\nu l}\sum_{i=1}^{l}\xi_i-\rho\\ s.t.\ & \left(\mathbf{v}\cdot\mathbf{x}_i\right)\geq\rho-\xi_i\\ & \xi_i\geq 0, i=1,\ldots,l\end{aligned} \tag{14}$$

It can be seen that the optimization problem (14) is similar in structure to standard one-class SVM. For solving (14), we consider its Lagrangian as:

$$L(\mathbf{u},\mathbf{v},\xi,\rho,\alpha,\beta)=\frac{1}{2}\mu_1\left(\mathbf{v}^T\mathbf{v}\right)+\frac{1}{\nu l}\sum_{i=1}^{l}\xi_i-\rho-\sum_{i=1}^{l}\alpha_i\left(\mathbf{v}^T\mathbf{x}_i-\rho+\xi_i\right)-\sum_{i=1}^{l}\beta_i\xi_i \tag{15}$$

According to Equations (11) to (13),

$$L(\mathbf{v},\xi,\rho,\alpha,\beta)=\frac{1}{2}\mu_1\left(\mathbf{v}^T\mathbf{v}\right)-\sum_{i=1}^{l}\alpha_i\mathbf{v}^T\mathbf{x}_i$$

$$= -\frac{1}{2\mu_1}\sum_{i,j=1}^{l}\alpha_i\alpha_j\left(\mathbf{x}_i\cdot\mathbf{x}_j\right) \tag{16}$$

Thus we can get the dual problem:

$$\begin{aligned}
&\min_{\alpha} \quad \frac{1}{2\mu_1}\sum_{i=1}^{l}\sum_{j=1}^{l}\alpha_i\alpha_j\left(\mathbf{x}_i\cdot\mathbf{x}_j\right)\\
&s.t. \quad 0\le\alpha_i\le\frac{1}{\nu l}\\
&\qquad \sum_{i=1}^{l}\alpha_i = 1, i=1,\ldots,l
\end{aligned} \tag{17}$$

Solving (17) determines the Lagrangian multipliers α_i^*, then we can get $\mathbf{v}$ and $\|\mathbf{v}\|^2$.

On the similar lines, we can let $\mu_2 = \|\mathbf{v}\|^2$ and $\mathbf{x}_j' = \mathbf{X}_j\mathbf{v}$. Then we can construct another optimal quadratic programming problem to solve $\mathbf{u}$ and $\|\mathbf{u}\|^2$:

$$\begin{aligned}
&\min_{\mathbf{u}\in\mathbb{R}^{n_2},\xi\in\mathbb{R}^{l},\rho\in\mathbb{R}} \quad \frac{1}{2}\mu_2\|\mathbf{u}\|^2+\frac{1}{\nu l}\sum_{j=1}^{l}\xi_j-\rho\\
&s.t. \quad \left(\mathbf{u}\cdot\mathbf{x}_j'\right)\ge\rho-\xi_j ?\\
&\qquad \xi_j\ge 0, j=1,\ldots,l
\end{aligned} \tag{18}$$

Thus, $\mathbf{u}$ and $\mathbf{v}$ can be obtained by iteratively solving the optimization problems (14) and (18). In our experiments, $\mathbf{u}$ is initially set to the vector of all ones. The steps involved in solving the linear one-class STM using the alternating projection method is summarized in Table 1.

Table1. Alternating Projection Algorithm for the Linear One-Class STM

Input: The training samples $\mathbf{X}_i\in\mathbb{R}^{n_1}\otimes\mathbb{R}^{n_2}$.

Output: The parameters in classification tensor plan $\mathbf{u}^*\in\mathbb{R}^{n_1}$, $\mathbf{v}^*\in\mathbb{R}^{n_2}$ and $\rho\in\mathbb{R}$.

Step 1Initialization: Let $\mathbf{u}=(1,\ldots,1)^T$.

Step 2 Obtain $\mathbf{v}$ by solving the optimization problems (14), where $\mu_1=\|\mathbf{u}\|^2$ and $\mathbf{x}_i=\mathbf{X}_i^T\mathbf{u}$.

Step 3 Obtain $\mathbf{u}$ by solving the optimization problems (18), where $\mu_2=\|\mathbf{v}\|^2$ and $\mathbf{x}_j'=\mathbf{X}_j\mathbf{v}$.

Step 4 Do step 2 and step 3 iteratively until convergence: If the iteration number exceeds the maximum number of iteration or all of the below convergence conditions are satisfied:

$$\|\mathbf{u}_i-\mathbf{u}_{i-1}\|\le tolerance$$

Step 5 End.

4. Experimental Evaluation

In this section, we evaluate the performance of linear one-class STM with experiments on 8 publicly available datasets from one-class classification datasets on David Tax's homepage [14]. These datasets are shown in Table 2, and the table also includes some considered target class in each dataset. In Table 2, n is the number of features (all scaled to [-1, 1]) in each dataset and m is the total number of data points. Since our algorithm considers tensor as input data, all these vector datasets are converted to corresponding tensor form, and the tensor size are shown in Table 2, as well.

Table 2. Information of the Various Dataset

Dataset	m	n	Tensor size	Target class name	Target samples
Iris	150	4	2*2	Virginica	50
Breast-Cancer	683	9	3*3	Benign	458
Heart	297	13	4*4	Absent	164
Imports	159	25	5*5	Large	88
Cancer-Wpbc	198	33	6*6	Non	151
Ionospheres	351	34	6*6	Good	225
Sonar	203	61	8*8	Rocks	97
Arrhythmia	420	278	17*17	Normal	237

The experiments results are compared with the standard linear one-class SVM. In all simulations, cross-validation is used on the training set to find the best parameter, and the possible choices for parameter is $\nu = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. All the simulations have been implemented in MATLAB R2011b on Windows 7 running on a PC with system configuration Intel Core i3 (2.4 GHz) and 6 GB of RAM.

4.1. Classification Performance

In this experiment, we compared the performance of the proposed linear one-class STM with the standard linear one-class SVM on all the publicly available datasets shown in Table 2. All the datasets are randomly split into training and testing sets. Since we particularly focus on the performance of small training size, we evaluate the results over 50 random splits and report the average performance of each algorithm.

Table 3, summarizes the average percentages of test accuracy of the two algorithms and the standard deviation of 50 times simulations as well. To evaluate the performance with respect to the training set size, we solve the two algorithms on various training sizes (2, 4, 6 and 8, training samples). The best results are bolded in each comparison. We can see that the linear one-class STM is better than the linear one-class SVM in 7 out of 8 datasets. In the remaining Sonar dataset, both algorithms are comparable. Especially when training set is small (2 training samples), linear one-class STM is more outstanding than linear one-class SVM in all 8 datasets. As the number of training samples increases, the average test accuracy of both algorithms tends to grow correspondingly.

Table 3. Average Percentages of Accuracy on Various Training Sample Sizes in All Datasets. I: The Linear One-Class STM; II: The Standard Linear Cne-Class SVM

Dataset	Alg.	Training sample size			
		2	4	6	8
Iris	I	**84.00**±8.31	**89.47**±6.78	**92.97**±5.28	**94.42**±3.92
	II	83.64±8.14	89.22±6.24	91.75±4.20	93.14±3.34
Breast-Cancer	I	**71.95**±16.85	**82.79**±11.67	**88.01**±7.89	**90.27**±5.61
	II	65.70±15.72	77.54±12.71	84.33±8.51	87.41±6.24
Heart	I	**54.96**±7.75	**62.59**±7.27	**67.37**±5.82	**69.27**±5.04
	II	50.72±3.82	57.65±5.58	62.85±5.93	66.84±5.75
Imports	I	**50.42**±10.18	**60.67**±8.65	**66.00**±7.74	**66.69**±6.31
	II	47.36±4.06	53.70±6.45	58.94±6.94	62.62±5.56
Cancer-Wpbc	I	**45.82**±10.47	**56.12**±8.41	**59.36**±6.51	**61.64**±6.30
	II	37.23±8.87	49.20±8.27	53.79±7.45	55.74±6.81
Ionospheres	I	**54.24**±12.98	**64.83**±10.50	**69.25**±8.21	**71.64**±7.01
	II	51.01±11.62	59.28±10.10	64.29±9.50	67.94±7.34
Sonar	I	**59.91**±4.13	61.32±3.91	60.51±4.41	60.13±4.27
	II	58.81±3.58	**62.14**±3.15	**61.82**±3.65	**61.57**±4.23
Arrhythmia	I	**58.61**±7.76	**64.39**±5.19	**65.52**±4.03	**66.05**±2.96
	II	47.70±3.93	56.89±6.37	60.98±5.78	63.13±4.50

The AUC, the area under the ROC curve, is always used to measure the performance of a one-class classifier [15]. Table 4, summarizes the average percentages of AUC of 50 times simulations on each dataset, and focuses on various small training sample sizes as well. The bolded numbers are the best average AUC of every pair comparison. We can see that in 6 out of 8 datasets, the linear one-class STM is better than the linear one-class SVM with the AUC. Furthermore, in the remaining 2, datasets, the AUCs of the two algorithms are much similar to each other.

Table 4. Average Percentages of AUC on Various Training Sample Sizes in All Datasets. I: The Linear One-Class STM; II: The Standard Linear One-Class SVM

Dataset	Alg.	Training sample size			
		2	4	6	8
Iris	I	**99.00**±1.60	**99.43**±0.89	**99.64**±0.67	**99.80**±0.15
	II	98.97±0.87	99.22±0.74	99.32±0.68	99.42±0.50
Breast-Cancer	I	**98.90**±2.45	**99.07**±1.08	**99.16**±0.91	99.15±0.91
	II	98.67±1.27	99.05±0.84	99.12±0.52	**99.18**±0.26
Heart	I	75.03±12.47	77.41±10.89	80.56±7.06	80.92±5.96
	II	**75.80**±12.06	**79.02**±8.98	**80.87**±8.23	**81.94**±6.16
Imports	I	64.39±22.48	**72.96**±12.63	**75.44**±8.39	**74.41**±7.45
	II	**65.70**±18.34	68.99±11.81	71.78±8.84	72.04±7.89
Cancer-Wpbc	I	**58.41**±2.94	**58.85**±2.69	**58.63**±2.70	**58.80**±2.80
	II	58.25±5.15	57.71±4.97	57.04±5.54	56.24±6.03
Ionospheres	I	**74.89**±7.36	**77.89**±7.24	**80.89**±6.61	**81.12**±5.96
	II	72.86±6.78	77.09±6.57	80.09±6.21	80.69±5.70
Sonar	I	**67.36**±5.81	**68.51**±4.55	**68.47**±4.81	68.22±4.77
	II	66.65±5.36	67.86±4.78	68.44±4.05	**68.38**±4.51
Arrhythmia	I	72.25±1.21	72.40±1.25	72.34±1.15	72.31±1.02
	II	**72.67**±1.76	**73.49**±1.33	**73.72**±1.18	**73.80**±1.13

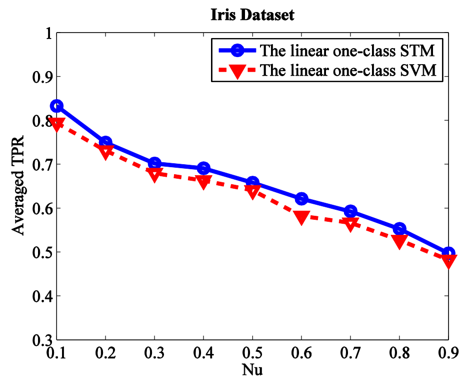
As we analysis above, both of these two evaluation indexes indicate that classifiers based on tensor representation are particularly suitable for small sample problems. This might be due to the fact that the number of parameters needs to be estimated in tensor classifiers is much smaller than in vector classifiers. As a summary, the linear one-class STM is outperforming the linear one-class SVM especially when training set is small.

4.2. Parameter Sensitivity

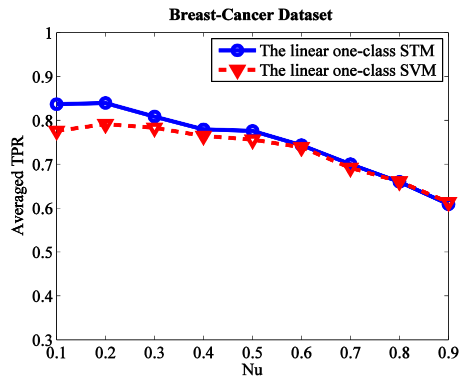
In this section, we discuss the classification performance of the two algorithms with special reference to parameter ν. In the standard one-class SVM algorithm, parameter ν bounds the fractions of outliers and support vectors from above and below [8]. Notice that now the training sets are pretty small since we are interested in small training size problems. Obviously, it is more meaningful to find the regularity corresponding with ν in tensor space rather than to validate how ν can be used to control the above fractions.

Now we consider the choices for parameter is $\nu = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. We train the linear one-class STM and linear one-class SVM with different parameter ν on random 8 samples training set. Without loss of generality, we evaluate the results over 50 times simulations and report the average performance of each algorithm. The experiment results indicate that the True Positive Rate (TPR) has significant relevance with parameter ν. To better understand, we illustrate the TPR with the two algorithms for each dataset, as shown in Figure 1.

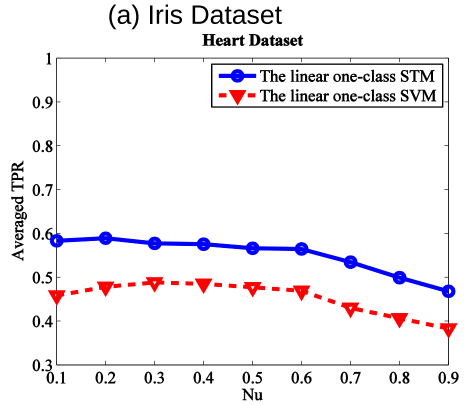
As shown in Figure 1, we can draw the conclusions from the above experimental results.



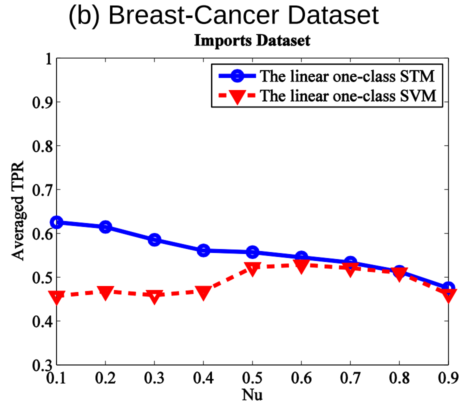
(a) Iris Dataset



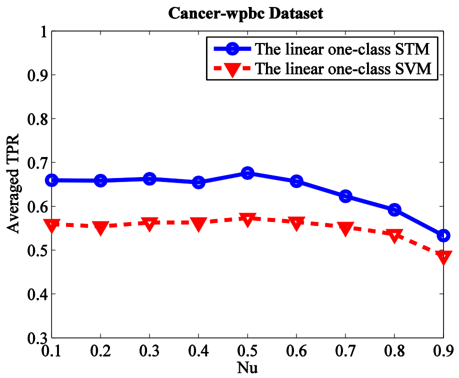
(b) Breast-Cancer Dataset



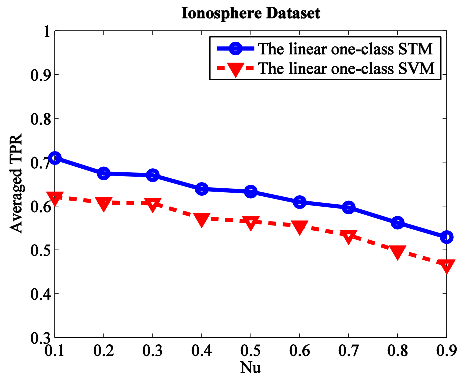
(c) Heart Dataset



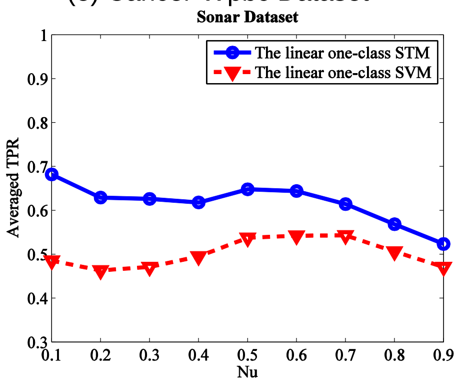
(d) Imports Dataset



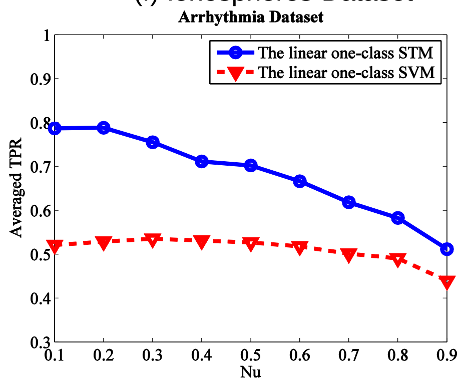
(e) Cancer-Wpbc Dataset



(f) Ionospheres Dataset



(g) Sonar Dataset



(h) Arrhythmia Dataset

Figure 1. Average TPR on Different Values of ν. The Comparisons of the Linear One-Class STM and The Linear One-Class SVM on Eight Datasets

(1) The TPRs of both the two algorithms tend to decrease with parameter ν .Especially for the linear one-class STM, the trend of decreasing is more obviously. In 5 out of 8 datasets, the best TPRs of tenser based algorithm appear when $\nu = 0.1$. In the remaining 3 datasets, the best TPRs appear when $\nu = 0.2$ or 0.5. However, when $\nu = 0.1$ the TPRs are closed to the best one, which are 0.7867 and 0.7882 in Arrhythmia dataset, 0.8369 and 0.8396 in Breast-Cancer dataset, and in Cancer-Wpbc dataset they are 0.6594 and 0.6759. In brief, we can conclude that parameter ν can control the maximal margin to the decision hyperplane in tensor space, which separates most target class samples from the origin.

(2) Tensor based algorithm has significant performance with TPR evaluation index. We can see that in all the 8 datasets, the TPRs of the linear one-class STM are much better than those of linear one-class SVM.

In short, the TPRs of the two algorithms have significant relevance with parameter ν . In some specific application fields, such as intrusion detection and outlier detection, only the normal labeled data can be collected. In such cases, the TPR is an available evaluation index. This experiment indicates that tensor based one-class classification algorithm is outstanding and the best parameter is also shown clearly in the analysis above.

5. Conclusions and Future Work

In this paper we have investigated a tensor-based one-class classification algorithm named linear one-class support tensor machine, which takes tensor as input data and learns a linear classifier in tensor space. The significant benefit of tensor representation is the reduction of the number of decision parameters, so that the new algorithm is especially suitable for small sample cases in high dimensional space. Our experimental results on publicly and available datasets demonstrate that tensor based classifiers are outperforming on small training size cases. We also discuss the parameter sensitivity in our tenser based one-class classifier. The experimental results show that the parameter v is concerned with the true positive rate of the classifiers.

In future work, we will investigate the kernel techniques of tensor data and generalize our algorithm to nonlinear cases. In this paper, we empirically study on the parameter sensitivity, and the theoretical guarantee is a promising direction on the work. Further study on this work will also include investigating more efficient computational methods for solving the optimization problems of linear one-class STM.

Acknowledgments

The work is supported by the National Science Foundation of China (Grant No. 11171346) and the "New Start" Academic Research Projections of Beijing Union University (Zk10201513).

References

[1] D. Cai, X. Hei and J. Ha, "Learning with tensor representation". Department of Computer Science, University of Illinois at Urbana-Champaign, UIUCDCS-R-2006-2716, **(2006)**.
[2] D. Tao, X. Li, X. Wu and S. J. Maybank, "General tensor discriminant analysis and gabor features for gait recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 10, **(2007)**, pp. 1700–1715.
[3] D. Tao, X. Li, X. Wu and S. J. Maybank, "Supervised tensor learning", Knowledge and Information Systems, vol. 13, no. 1, **(2007)**, pp. 1–42.
[4] D. Cai, X. He, J. Wen, J. Han and W. Ma, "Support tensor machines for text categorization", Technical report, Computer Science Department, UIUCDCS-R-200-2714, (**2006**).
[5] X. Zhang, X. Gao and Y. Wang, "Twin support tensor machines for MCs detection", Journal of Electronics (China), vol. 26, no, 3, **(2009)**, pp. 318-325.
[6] R. Khemchandani, A. Karpatne and S. Chandra, "Proximal support tensor machines", Int. J. Mach. Learn. & Cyber. 4:703-712, **(2013)**.
[7] Z. Hao, L. He, B. Chen and X. Yang, "A linear support higher-order tensor machine for classification", IEEE Transactions on Image Processing, vol. 22, no. 7, **(2013)**, pp. 2911-2920.
[8] B. Scholkopf, J. C. Platt, J. S. Taylor, A. J. Smola and R. C. Williamson, "Estimating the support of a high-dimensional distribution", Technical report, Microsoft Research, MSR-TR-99-87, **(1999)**.
[9] D. M. J. Tax and R. P. W. Duin, "Support vector domain description", Pattern Recognition Letter, vol. 20, no. 11, **(1999)**, pp. 1191–1199.
[10] L. M. Manevitz and M. Yousef, "One-Class SVMs for Document Classification", Journal of Machine Learning Research, vol. 2, **(2001)**, pp. 139-154.
[11] Y. Chen, X. Zhou and T. S. Huang, "One-class SVM for learning in image retrieval", In Proc. IEEE Int'l Conf. on Image Processing, Thessaloniki, Greece, **(2001)**.
[12] W. Zhu and P. Zhong, "A new one-class SVM based on hidden information", Knowledge-Based Systems, vol. 60, **(2014)**, pp. 35–43.
[13] Z. Zhao, P. Zhong and Y. Zhao, "Reduced least squares one-class SVM in empirical feature space for imbalanced data", ICIC Express Letters, vol. 5, no. 11, **(2010)**, pp. 4115-4121.
[14] http://prlab.tudelft.nl/users/david-tax.
[15] A. Airola, T. Pahikkala, W. Waegeman, B. D. Baets and T. Salakoski, "An experimental comparison of cross-validation techniques for estimating the area under the ROC curve", Comput.Stat. Data Anal., vol. 55, no. 4, (**2011**), pp. 1828–1844.

