Vietnam Journal of Computer Science

Twin Support Vector Machine with Multi-Clusters Data

--Manuscript Draft--

Manuscript Number:	WSPC-VJCS-D-24-00193R1
Full Title:	Twin Support Vector Machine with Multi-Clusters Data
Article Type:	Research Paper
Keywords:	Support Vector Machine; Twin Support Vector Machine; Structural Twin Support Vector Machine; Structural granularity.
Abstract:	With the rapid development of data, data sets are increasing in number and diversifying in structure. In binary classification problems, two data classes seem to be more complicated. Each class may be constructing by more than one cluster, and the number of data points of clusters being different (we call multi-clusters data). Traditional algorithms such as Support Vector Machines (SVM), together two typical innovations of SVM are Twin Support Vector Machine (TSVM) and Structural Twin Support Vector Machine (S-TSVM) don't sufficiently exploit data information: structural information with cluster granularity and information about the number of data points in each cluster. This may affect the accuracy of binary classification problems. This paper proposes a new method to deal with binary classification problems for multi-clusters data, a Twin Support Vector Machine with Multi-Clusters Data (called TSVM-MCD) using a cluster-vs-class strategy. Don't like TSVM and S-TSVM, TSVM-MCD exploits structural information with cluster granularity of data training, and the information about the number of data points of each cluster. Both theorized and experimentally, we show the detail comparison of TSVM-MCD with TSVM and S-TSVM.
Response to Reviewers:	Reply to reviewer Paper title: Twin Support Vector Machine with Multi-Clusters Data Ms. No. WSPC-VJCS-D-24-00193 Authors We would like to thank the referees for their helpful comments and valuable suggestions. The paper has been revised according to their requirements (the changes are highlighted in the article). The detailed modifications are listed below. 1. The last paragraph of the Introduction section should be relocated to the Background section for improved organization and clarity. R: We have relocated the last paragraph of the Introduction section as suggestion. 2. In the abstract, the statement "Both theorized and experimentally, we show the comparison of TSVM-MCD with two improvements of SVM" lacks a theoretical comparison of the algorithms, as indicated. 3. The abstract needs rephrasing to better highlight the originality and contribution of the manuscript beyond just comparing algorithms. R: They are true that, so we have rewritten the abstract and added a discussion section to take a more detailed look at the algorithms. 4. Meaningful results should be thoroughly explained, particularly instances where the performance of algorithms is equal. Providing insights into why this occurred can enhance the manuscript's comprehensibility. R: We have added more explained for the performance of algorithms. 5. The rationale behind selecting the two improved SVM algorithms should be elucidated. Additionally, clarification is needed regarding any significance associated with the proposals of Structural Twin Support Vector Machine in 2013 and Twin

	Support Vector Machines in 2007. R: We have more explained for these two algorithms. 6. In the introduction section, the authors should mention the main contributions and effectiveness of the proposed method. 7. In the introduction section, the authors should mention the main contributions and effectiveness of the proposed method. R: We have added these suggestions in the Introduction section. 8. The experimental data is from only one source. The authors may consider adding more data sources to demonstrate the effectiveness of the algorithm. The authors consider further comparisons with other algorithms such as LS-TWSVM (the reviewer mean that Least Square Twin Support Vector Machine). R: We know more difference data source. However in this paper, we introduces a novel approach for addressing binary classification problems, and in the original results they used these data sets. We also know more new research that are related to TSVM and S-TSVM, but they not closed to the novel approach of TSVM and S-TSVM. For example, the LS-TSVM has the same approach, but it find the global solution by using "least square" method, while TSVM, S-TSVM and TSVM-MCD solving the Quadratic Programming Problem to find the optimal solution. 9. Why did the author not compare with the author's previous proposed method HM-TWSVM (mean that Hierarchical Multi Twin Support Vector Machine). R: This is the first result when we study on SVMs. That is not a good ideal, because we thought that all clusters in each class have the same role in the classification problem. We have added this paper in the References section, and explained why in the Introduction. Thank so much Authors

Vietnam Journal of Computer Science

Twin Support Vector Machine with Multi-Clusters Data

The Cuong Nguyen*

Faculty of Basic, Telecommunications University, Mai Xuan Thuong
Nha Trang, Khanh Hoa, Vietnam
thecuong@tcu.edu.vn

Van Han Nguyen

Faculty of Information Technology, Nguyen Tat Thanh University
Ho Chi Minh City, Vietnam

Received (Day Month Year)
Revised (Day Month Year)

With the rapid development of data, data sets are increasing in number and diversifying in structure. In binary classification problems, two data classes seem to be more complicated. Each class may be constructing by more than one cluster, and the number of data points of clusters being different (we call multi-clusters data). Traditional algorithms such as Support Vector Machines (SVM), together two typical innovations of SVM are Twin Support Vector Machine (TSVM) and Structural Twin Support Vector Machine (S-TSVM) don't sufficiently exploit data information: structural information with cluster granularity and information about the number of data points in each cluster. This may affect the accuracy of binary classification problems. This paper proposes a new method to deal with binary classification problems for multi-clusters data, a Twin Support Vector Machine with Multi-Clusters Data (called TSVM-MCD) using a cluster-vs-class strategy. Don't like TSVM and S-TSVM, TSVM-MCD exploits structural information with cluster granularity of data training, and the information about the number of data points of each cluster. Both theorized and experimentally, we show the detail comparison of TSVM-MCD with TSVM and S-TSVM.

Keywords: Support Vector Machine, Twin Support Vector Machine, Structural Twin Support Vector Machine, Structural granularity.

1. Introduction

In the latter half of the 20th century, the issue of classifying data into two categories was first introduced and explored. However, the data sets collected during that time were relatively simple. As time passed, real-world data became more diverse in structure and quantity. Different clusters often establish data sets, each cluster has one distributed trend and the number of data points of each cluster is different (multi-clusters data). For example, we consider the problem of classifying fruits with

*Corresponding author.

data including five categories: Mango, Jackfruit, Pineapple, Apples, and Grapes, but the fruits will only be classified according to the criteria "smooth skin" or "rough skin". Data in the "smooth skin" class will form 3 clusters, corresponding to Mango, Apples, and Grapes. In comparison, data in the "rough skin" class will be distributed into 2 clusters, corresponding to Jackfruit and Pineapple.

As a result, binary classification algorithms also needed to be improved to better handle the increasing diversity of data. Support Vector Machine (SVM)[1,2] was a popular binary classification algorithm applied to many different fields in practice.[3–7] The main idea of SVM is to find a hyperplane separating two classes with the largest margin. However, SVM does not fully exploit structural information and information about the number of data points of clusters. Many variants of SVM have been recently proposed to improve the accuracy and other tasks of standard SVM.[6–10] Two typical innovations of SVM are the Twin Support Vector Machine (TSVM),[11] and the Structural Twin Support Vector Machine (S-TSVM).[12] The main idea of TSVM is to seek two hyperplanes such that each hyperplane is closer to one class and put the remaining class to one side by solving two Quadratic Programming Problems (QPPs)[13] whose sizes are smaller than the QPP in SVM. S-TSVM has the same strategy as TSVM. Besides, S-TSVM fully exploits structural information with cluster granularity into learning the model to build a more reasonable classifier. However, both TSVM and S-TSVM don't use information about the number of data points in each cluster. Recently, we publish Hierarchical Multi Twin Support Vector Machine (HM-TWSVM)[14] with cluster-vs-class strategy. The HM-TWSVM exploits structural information with cluster granularity, but it considers clusters in each class to have the same role in binary classification problem, that's not the good ideal.

Based on the strategy of TSVM[11] and S-TSVM,[12] we propose a new binary classification model: Twin Support Vector Machine with Multi-Clusters Data (called TSVM-MCD) with a cluster-vs-class strategy. Instead of solving two QPPs as in S-TSVM, TSVM-MCD will solve $(k + l)$ QPPs, where k and l are the number of clusters in class $\{+\}$ and class $\{-\}$, respectively. This method allows TSVM-MCD effectively describe the distribution trend of each cluster in each class, so its ability to generalize data is better and may improve classification accuracy for binary classification problems. Especially for multi-clusters data, TSVM-MCD sufficiently exploiting information about the structure with cluster granularity, and the number of data points in each cluster of two classes to train model.

The paper is organized as follows: Section 2 briefly introduces the background of QPP, Structural granularity, SVM, TSVM, and S-TSVM; Section 3 is a detailed description of TSVM-MCD along with the algorithms and discussions; All experimental results are shown in Section 4, together with the comparative evaluation; The conclusion is given in Section 5. All algorithms are settled by Python Programming Language.

2. Background

In this paper, all real numbers are denoted by normal letters, all vectors will be column vectors and denoted by bold letters, transformed to a row vector by ^{T}. A column vector of ones in real space of arbitrary dimension will be denoted by $\mathbf{e}$. All matrices will be denoted by bold capital letters. The identity matrix of arbitrary dimension will be denoted by $\mathbf{I}$, and $\|\|$ is the Euclidean norm.

In this section, we first briefly describe the background of QPP, Structural granularity, SVM,[1] TSVM,[11] and S-TSVM.[12]

2.1. *The Quadratic Programming Problem.*

The general form of the problem is as follows:

$$\text{QPP}: \begin{cases} Q(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{G}\mathbf{x} + \mathbf{g}^T\mathbf{x} + \alpha \longrightarrow \min, \\ \mathbf{p}_i^T\mathbf{x} \geq b_i, \quad i \in I := \{1, \dots, p\}, \\ \mathbf{q}_j^T\mathbf{x} = d_j, \quad j \in J := \{1, \dots, q\}. \end{cases}$$

By setting $\mathbf{P} = [\mathbf{p}_1, \dots, \mathbf{p}_p]^T$ is the matrix consisting of p row vectors $\mathbf{p}_i^T$, $\mathbf{Q} = [\mathbf{q}_1, \dots, \mathbf{q}_q]^T$ is the matrix consisting of q row vectors $\mathbf{q}_j^T$, $\mathbf{b} = (b_1, b_2, \dots, b_p)$, $\mathbf{d} = (d_1, d_2, \dots, d_q)$.

The matrix form of the QPP is as follows:

$$\text{QPP}: \begin{cases} Q(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{G}\mathbf{x} + \mathbf{g}^T\mathbf{x} + \alpha \longrightarrow \min, \\ \mathbf{P}\mathbf{x} \geq \mathbf{b}, \\ \mathbf{Q}\mathbf{x} = \mathbf{d}. \end{cases}$$

Where $\mathbf{G} \in \mathbb{R}^{n\times n}$ is the symmetric matrix, $\mathbf{P} \in \mathbb{R}^{p\times n}$, $\mathbf{Q} \in \mathbb{R}^{q\times n}$, $\mathbf{b} \in \mathbb{R}^p$, $\mathbf{d} \in \mathbb{R}^q$, $\mathbf{g}, \mathbf{x} \in \mathbb{R}^n$, $\alpha \in \mathbb{R}$. When the objective function Q is convex (i.e. $\mathbf{G}$ is positive semi-definite), the QPP is a convex problem.

Theorem 2.1 (Optimal conditions (see Ref. 13, pp. 412)).

(a) Suppose that $\mathbf{x}^*$ is the solution of QPP. Then there are coefficients $\boldsymbol{\lambda}^* = (\lambda_1^*, \dots, \lambda_p^*) \in \mathbb{R}^p$, $\boldsymbol{\mu}^* = (\mu_1^*, \dots, \mu_q^*) \in \mathbb{R}^q$ satisfying the following conditions:

$$\begin{cases} \mathbf{G}\mathbf{x}^* + \mathbf{g} = \sum_{i=1}^{p} \lambda_i^* \mathbf{p}_i + \sum_{j=1}^{q} \mu_j^* \mathbf{q}_j, & \\ \mathbf{p}_i^T\mathbf{x}^* \geq b_i,\ \lambda_i^* \geq 0, & i \in I = \{1, \dots, p\}, \\ \lambda_i^*(\mathbf{p}_i^T\mathbf{x}^* - b_i) = 0, & i \in I, \\ \mathbf{q}_j^T\mathbf{x}^* = d_j, & j \in J = \{1, \dots, q\}. \end{cases}$$

The above conditions are called the KKT system $(Karush - Kuhn - Tucker)$ of QPP, $\mathbf{x}^*$ is called a KKT point, and the coefficients $\boldsymbol{\lambda}^*$, $\boldsymbol{\mu}^*$ are called Lagrange multipliers corresponding to $\mathbf{x}^*$.

(b) If the QPP is convex, and $\mathbf{x}^*$ is a KKT point with Lagrange multipliers $\boldsymbol{\lambda}^*$, $\boldsymbol{\mu}^*$, then $\mathbf{x}^*$ is also a solution of QPP.

Note that, the KKT system can be rewritten in the matrix form as follows:

$$\begin{cases} \mathbf{G}\mathbf{x}^* + \mathbf{g} = \mathbf{P}^T\boldsymbol{\lambda}^* + \mathbf{Q}^T\boldsymbol{\mu}^*, \\ \mathbf{P}\mathbf{x}^* \geq \mathbf{b},\ \boldsymbol{\lambda}^* \geq \mathbf{0}, \\ \boldsymbol{\lambda}^{*T}(\mathbf{P}\mathbf{x}^* - \mathbf{b}) = \mathbf{0}, \\ \mathbf{Q}\mathbf{x}^* = \mathbf{d}. \end{cases}$$

2.2. *Structural granularity*

Consider a binary classification problem with the multi-clusters data set, denoted by a matrix $\mathbf{C}$, consisting of m points (each point is a row of $\mathbf{C}$) $\mathbf{x}_j^T \in \mathbb{R}^n$, $1 \leq j \leq m$. We also write $\mathbf{x}_j \in \mathbf{C}$ to indicate that $\mathbf{x}_j$ is a row of $\mathbf{C}$. Suppose that $y_j \in \{-1, 1\}$ is the $j-$th data point label corresponding to $\mathbf{x}_j$. Class $\{+\}$ consists of m_A points denoted by a matrix $\mathbf{A} \subset \mathbb{R}^{m_A \times n}$, and class $\{-\}$ consists of m_B points denoted by a matrix $\mathbf{B} \subset \mathbb{R}^{m_B \times n}$. There are k clusters in class $\mathbf{A}$, whose $i-$th cluster consists of m_{Ai} points and is denoted by matrix $\mathbf{A}_i \subset \mathbb{R}^{m_{Ai} \times n}, i = 1, \ldots, k$. Also, there are l clusters in class $\mathbf{B}$, whose $j-$th cluster consists of m_{Bj} points and is denoted by matrix $\mathbf{B}_j \subset \mathbb{R}^{m_{Bj} \times n}, j = 1, \ldots, l$. Here, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{A}_i$, $\mathbf{B}_j$ are called structural granularity.[15] We are interested in the following quantities of structural granularity.

- Class granularity:
  $\varSigma_A = \frac{1}{m_A} \sum\limits_{\mathbf{x}_j \in \mathbf{A}} (\mathbf{x}_j - \boldsymbol{\mu}_A)(\mathbf{x}_j - \boldsymbol{\mu}_A)^T,$
  $\varSigma_B = \frac{1}{m_B} \sum\limits_{\mathbf{x}_j \in \mathbf{B}} (\mathbf{x}_j - \boldsymbol{\mu}_B)(\mathbf{x}_j - \boldsymbol{\mu}_B)^T.$
- Cluster granularity:
  $\varSigma_{Ai} = \frac{1}{m_{Ai}} \sum\limits_{\mathbf{x}_j \in \mathbf{A}_i} (\mathbf{x}_j - \boldsymbol{\mu}_{Ai})(\mathbf{x}_j - \boldsymbol{\mu}_{Ai})^T,$
  $\varSigma_{Bj} = \frac{1}{m_{Bj}} \sum\limits_{\mathbf{x}_j \in \mathbf{B}_j} (\mathbf{x}_j - \boldsymbol{\mu}_{Bj})(\mathbf{x}_j - \boldsymbol{\mu}_{Bj})^T.$

Here, $\boldsymbol{\mu}_X$ denotes the average vector of the data set $\mathbf{X}$. When the data set is standardized we have $\varSigma_X = \frac{1}{m_X}\mathbf{X}^T\mathbf{X}$

2.3. *SVM, TSVM, S-TSVM*

The main idea of Support Vector Machines (SVM)[1] is to seek a hyperplane

$$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b = 0, \quad \mathbf{w} \in \mathbb{R}^n, b \in \mathbb{R},$$

which separates class $\mathbf{A}$ and class $\mathbf{B}$ such that the margin $\frac{2}{\|\mathbf{w}\|}$ between the two classes is the largest, by solving the QPP as follows:

$$\begin{cases} \min\limits_{\mathbf{w},\ b,\ \boldsymbol{\xi}} & c\mathbf{e}^T\boldsymbol{\xi} + \frac{1}{2}\|\mathbf{w}\|^2, \\ \text{s.t.} & \mathbf{D}(\mathbf{C}\mathbf{w} + \mathbf{e}b) + \boldsymbol{\xi} \geq \mathbf{e},\ \boldsymbol{\xi} \geq \mathbf{0}, \end{cases} \tag{1}$$

A new data point $\mathbf{x}$ will be classified in class $\mathbf{A}$ if $\text{sgn}(f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b) > 0$ and in class $\mathbf{B}$ if $\text{sgn}(f(\mathbf{x})) < 0$.

The main idea of Twin Support Vector Machines (TSVM)[11] is to seek two hyperplanes:

- $f_+(\mathbf{x})(=\mathbf{w}_+^T\mathbf{x}+b_+)=0$ is closer to class $\mathbf{A}$ and far away from class $\mathbf{B}$,
- $f_-(\mathbf{x})(=\mathbf{w}_-^T\mathbf{x}+b_-)=0$ is closer to class $\mathbf{B}$ and far away from class $\mathbf{A}$,

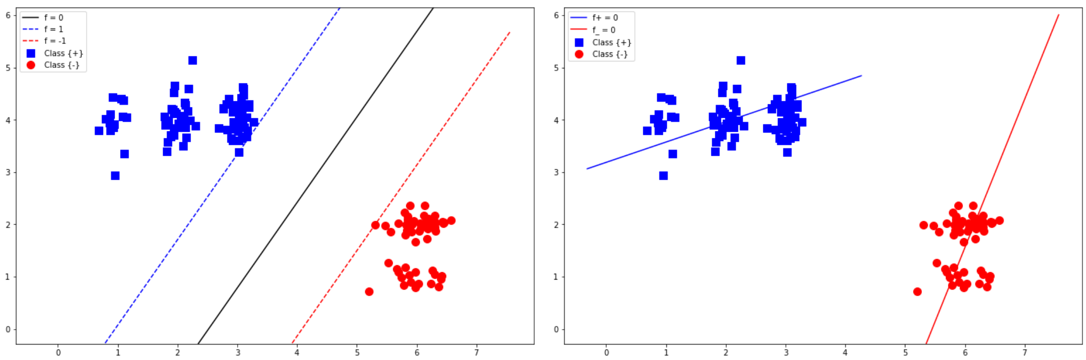
by solving two QPPs as follows:

$$\begin{cases} \min\limits_{\mathbf{w}_+,b_+,\boldsymbol{\xi}} & \frac{1}{2}\|\mathbf{A}\mathbf{w}_+ + \mathbf{e}_A b_+\|^2 + c_1\mathbf{e}_B^T\boldsymbol{\xi}, \\ \text{s.t.} & -(\mathbf{B}\mathbf{w}_+ + \mathbf{e}_B b_+) + \boldsymbol{\xi} \geq \mathbf{e}_B,\ \boldsymbol{\xi} \geq \mathbf{0}, \end{cases} \tag{2}$$

and

$$\begin{cases} \min\limits_{\mathbf{w}_-,b_-,\boldsymbol{\eta}} & \frac{1}{2}||\mathbf{B}\mathbf{w}_- + \mathbf{e}_B b_-\|^2 + c_2\mathbf{e}_A^T\boldsymbol{\eta}, \\ \text{s.t.} & (\mathbf{A}\mathbf{w}_+ + \mathbf{e}_A b_+) + \boldsymbol{\eta} \geq \mathbf{e}_A,\ \boldsymbol{\eta} \geq \mathbf{0}. \end{cases} \tag{3}$$

Where $c_1, c_2 \in \mathbb{R}$ are penalty coefficients, $\mathbf{e}_A \in \mathbb{R}^{m_A}$, $\mathbf{e}_B \in \mathbb{R}^{m_B}$ are vectors of ones, $\boldsymbol{\xi} \in \mathbb{R}^{m_B}$, $\boldsymbol{\eta} \in \mathbb{R}^{m_A}$ are vectors of slack variables. A new data $\mathbf{x}$ is classified to class $\mathbf{A}$ or class $\mathbf{B}$ depending on whether it is closer to the hyperplane $f_+(\mathbf{x}) = 0$ or $f_-(\mathbf{x}) = 0$. The SVM and TSVM do not sufficiently exploit structural information with cluster granularity of data, so they describe the distribution structure of two classes being not exactly (see Figure 1).



(a) (b)

Fig. 1. The simple case, in which clusters in each class are constituted by the same trend, the SVM, and TSVM do not exactly describe the distribution trend of data in each class, and don't use information about the number of data points of clusters in each class. (a) SVM, (b) TSVM.

Structural Twin Support Vector Machine (S-TSVM)[12] has two steps: The first step is to extract the structural information within classes. The second step is the model learning. S-TSVM also determines two hyperplanes as TSVM:

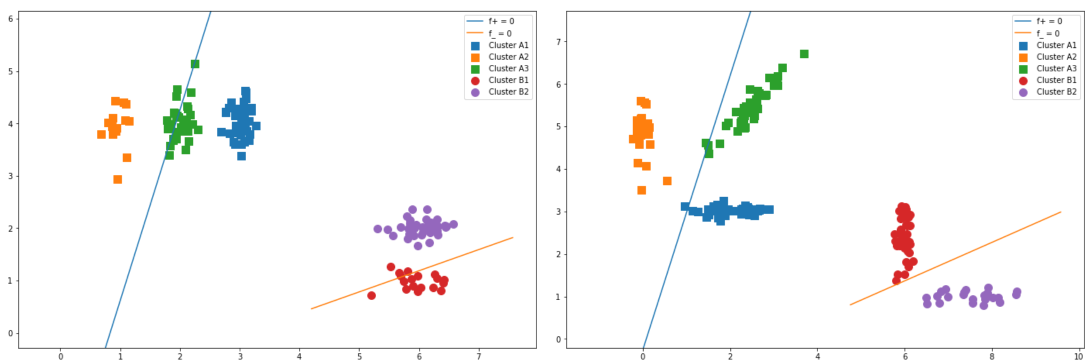
$$f_+(\mathbf{x}) = \mathbf{w}_+^T\mathbf{x} + b_+ = 0;\ f_-(\mathbf{x}) = \mathbf{w}_-^T\mathbf{x} + b_- = 0, \tag{4}$$

by solving two QPPs as follows:

$$\begin{cases} \min\limits_{\mathbf{w}_+, b_+, \boldsymbol{\xi}} & \frac{1}{2}\|\mathbf{A}\mathbf{w}_+ + \mathbf{e}_A b_+\|^2 + c_1 \mathbf{e}_B^T \boldsymbol{\xi} + \frac{1}{2}c_2(\|\mathbf{w}_+\|^2 + b_+^2) + \frac{1}{2}c_3 \mathbf{w}_+^T \varSigma_+ \mathbf{w}_+, \\ \text{s.t.} & -(\mathbf{B}\mathbf{w}_+ + \mathbf{e}_B b_+) + \boldsymbol{\xi} \geq \mathbf{e}_B,\ \boldsymbol{\xi} \geq \mathbf{0}, \end{cases} \tag{5}$$

$$\begin{cases} \min\limits_{\mathbf{w}_-, b_-, \boldsymbol{\eta}} & \frac{1}{2}\|\mathbf{B}\mathbf{w}_- + \mathbf{e}_B b_-\|^2 + c_4 \mathbf{e}_A^T \boldsymbol{\eta} + \frac{1}{2}c_5(\|\mathbf{w}_-\|^2 + b_-^2) + \frac{1}{2}c_6 \mathbf{w}_-^T \varSigma_- \mathbf{w}_-, \\ \text{s.t.} & (\mathbf{A}\mathbf{w}_- + \mathbf{e}_A b_-) + \boldsymbol{\eta} \geq \mathbf{e}_A, \boldsymbol{\eta} \geq \mathbf{0}. \end{cases} \tag{6}$$

Where $\varSigma_+ = \varSigma_{1+} + \cdots + \varSigma_{k+}$, $\varSigma_- = \varSigma_{1-} + \cdots + \varSigma_{l-}$, $\varSigma_{i+}$ and $\varSigma_{j-}$ are respectively the covariance matrices corresponding to the clusters $\mathbf{A}_i$ and $\mathbf{B}_j$, $\mathbf{e}_A \in \mathbb{R}^{m_A}$, $\mathbf{e}_B \in \mathbb{R}^{m_B}$ are vectors of ones. In problem (5), $\frac{1}{2}\|\mathbf{A}\mathbf{w}_+ + \mathbf{e}_A b_+\|^2$ is the sum of the squares of the distances from data points in class $\mathbf{A}$ to the hyperplane $\{f_+(\mathbf{x}) = 0\}$. $c_1 \mathbf{e}_B^T \boldsymbol{\xi}$ is the sum of errors, $\frac{1}{2}c_2(\|\mathbf{w}_+\|^2 + b_+^2)$ is the regularization, $\frac{1}{2}c_3 \mathbf{w}_+^T \varSigma_+ \mathbf{w}_+$ is the sum of covariance matrices with the cluster granularity of class $\mathbf{A}$ projected onto vector $\mathbf{w}_+$. The problem (6) is similarly established for class $\mathbf{B}$. A new data point is assigned to class $\mathbf{A}$ or class $\mathbf{B}$ in the same manner as in TSVM. The S-TSVM exploits structural information with cluster granularity of one class in each problem. Therefore, the ability of S-TSVM to describe data trends is more accurate than that of TSVM (see Figure 2 in the simple case). However, when data becomes more complex, the ability to describe data trends of the S-TSVM remains limited (see Figure 2 in complex case).



(c) (d)

Fig. 2. (c) In the simple case, S-TSVM describes quite accurately the distribution trend in each class. (d) In complex cases, S-TSVM has difficulty in describing the distribution trend of data and doesn't use information about the number of data points of clusters in each class.

3. TSVM with Multi-Clusters Data

This section introduces a new method to solve binary classification problems with multi-clusters data: Twin Support Vector Machine with Multi-Clusters Data (called

TSVM-MCD). Similar to the S-TSVM,[12] TSVM-MCD also has two steps. The first step is to group data in each class by Ward's linkage clustering method;[16] the second step is model learning. Suppose, there are k clusters in class $\mathbf{A}$, and l clusters in class $\mathbf{B}$. TSVM-MCD uses a cluster-vs-class strategy to determine $(k+l)$ hyperplanes such that each of which is closer to one cluster and far away from the other class. Specifically, the method needs to find k hyperplanes such that the $i-$th hyperplane, $f_{i+}(\mathbf{x}) = \mathbf{w}_{i+}^T\mathbf{x} + b_{i+} = 0, i = 1, \ldots, k$, is closer to cluster $\mathbf{A}_i$ and far away from class $\mathbf{B}$; Also, it needs to find l hyperplanes such that the $j-$th hyperplane, $f_{j-}(\mathbf{x}) = \mathbf{w}_{j-}^T\mathbf{x} + b_{j-} = 0, j = 1, \ldots, l$, is closer to cluster $\mathbf{B}_j$ and far away from class $\mathbf{A}$ (see Figure 3); Here $\mathbf{w}_{i+}, \mathbf{w}_{j-} \in \mathbb{R}^n, b_{i+}, b_{j-} \in \mathbb{R}$.

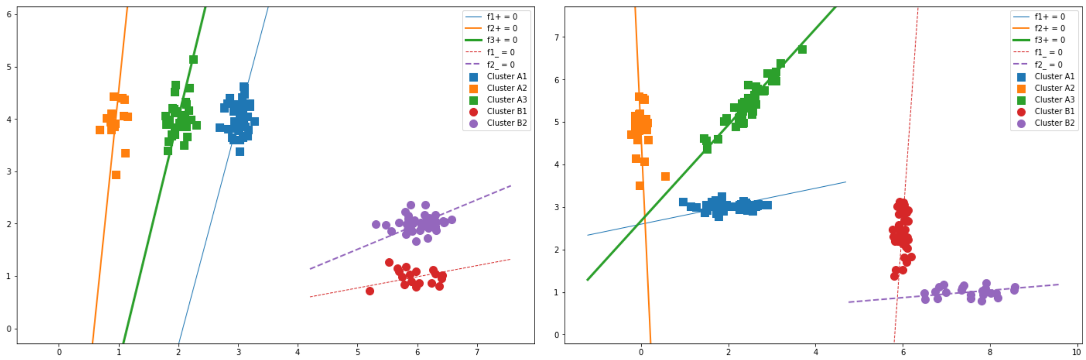
The classifier is now selected as:

$$f(\mathbf{x}) = \underset{+,\ -}{\text{argmin}}(f_+(\mathbf{x}),\ f_-(\mathbf{x})), \tag{7}$$

with

$$f_+(\mathbf{x}) = \sum_{i=1}^{k} \frac{m_{Ai}}{m_A} f_{i+}(\mathbf{x});\ \ f_-(\mathbf{x}) = \sum_{j=1}^{l} \frac{m_{Bj}}{m_B} f_{j-}(\mathbf{x}). \tag{8}$$

From (8), we can see that $f_+(\mathbf{x})$ is the weighted average of the distances from $\mathbf{x}$ to the hyperplanes $\{f_{i+}(\mathbf{x}) = 0\}$. The $i-$th hyperplane's weight is proportional to m_{Ai} - the number of data points in the cluster $\mathbf{A}i$. Similarly, $f_-(\mathbf{x})$ is the weighted average of distances from $\mathbf{x}$ to the hyperplanes $\{f_{j-}(\mathbf{x}) = 0\}$. By (7), a new data point $\mathbf{x}$ is classified into class $\mathbf{A}$ or $\mathbf{B}$ depending on whether $f_+(\mathbf{x})$ is less than or greater than $f_-(\mathbf{x})$.



(e) (f)

Fig. 3. The TSVM-MCD exploits structural information with cluster granularity, and information about the number of data points of clusters in each class to model learning. Consequently, the TSVM-MCD can describe the distribution trends of each cluster in each class, so the generalized capability is better than the S-TSVM and TSVM methods. (e) TSVM-MCD in simple case. (f) TSVM-MCD in complex case.

3.1. *The linear case*

When two classes are nearly linear separable, we determine $(k+l)$ hyperplanes in TSVM-MCD by solving $(k+l)$ QPPs as follows:

$$\begin{cases} \min\limits_{\mathbf{w}_{i+}, b_{i+}, \boldsymbol{\xi}} & \frac{1}{2}\|\mathbf{A}_i\mathbf{w}_{i+} + \mathbf{e}_{Ai}b_{i+}\|^2 + \frac{1}{2}c_1\mathbf{e}_B^T\boldsymbol{\xi} + \frac{1}{2}c_2(\|\mathbf{w}_{i+}\|^2 + b_{i+}^2), \\ \text{s.t.} & (\mathbf{B}\mathbf{w}_{i+} + \mathbf{e}_B b_{i+}) + \boldsymbol{\xi} \geq \mathbf{e}_B, \boldsymbol{\xi} \geq \mathbf{0}, \end{cases} \tag{9}$$

$i = 1, \ldots,\ k$ and

$$\begin{cases} \min\limits_{\mathbf{w}_{j-}, b_{j-}, \boldsymbol{\eta}} & \frac{1}{2}\|\mathbf{B}_j\mathbf{w}_{j-} + \mathbf{e}_{Bj}b_{j-}\|^2 + \frac{1}{2}c_3\mathbf{e}_A^T\boldsymbol{\eta} + \frac{1}{2}c_4(\|\mathbf{w}_{j-}\|^2 + b_{j-}^2), \\ \text{s.t.} & (\mathbf{A}\mathbf{w}_{j-} + \mathbf{e}_A b_{j-}) + \boldsymbol{\eta} \geq \mathbf{e}_A, \boldsymbol{\eta} \geq \mathbf{0}, \end{cases} \tag{10}$$

$j = 1, \ldots,\ l.$

Here, $\mathbf{e}_{Ai} \in \mathbb{R}^{m_{Ai}\times 1}$, $\mathbf{e}_{Bj} \in \mathbb{R}^{m_{Bj}\times 1}$, $\mathbf{e}_A \in \mathbb{R}^{m_A\times 1}$, and $\mathbf{e}_B \in \mathbb{R}^{m_B\times 1}$ are vectors of ones. $\boldsymbol{\eta} \in \mathbb{R}^{m_A\times 1}$, and $\boldsymbol{\xi} \in \mathbb{R}^{m_B\times 1}$ are vectors of slack variables. c_1, c_2, c_3, c_4 are penalty coefficients. In the problem (9), $\|\mathbf{A}_i\mathbf{w}_{i+} + \mathbf{e}_{Ai}b_{i+}\|^2$ is the sum of squares of distances from data points in cluster $\mathbf{A}_i$ to the hyperplane $f_{i+}(\mathbf{x}) = 0$, and it takes structural information with cluster granularity. Therefore, we do not need to add the $\frac{1}{2}c_3\mathbf{w}_+^T\Sigma_+\mathbf{w}_+$ term as in S-TSVM. $\frac{1}{2}c_1\mathbf{e}_B^T\boldsymbol{\xi}$ is the sum of errors, $\frac{1}{2}c_2(\|\mathbf{w}_{i+}\|^2 + b_{i+}^2)$ is the regularization term. The constraints of problem (9) are defined by the points of class $\mathbf{B}$. The problem (10) is similarly established for clusters $\mathbf{B}_j$ of class $\mathbf{B}$ with the constraints defined by class $\mathbf{A}$.

We will solve two problems (9), and (10) by solving two dual problems. Specifically, the Lagrange function of (9) is given by:

$$\mathcal{L}(\mathbf{w}_{i+}, b_{i+}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\|\mathbf{A}_i\mathbf{w}_{i+} + \mathbf{e}_{Ai}b_{i+}\|^2 + c_1\mathbf{e}_B^T\boldsymbol{\xi} + \\ \frac{1}{2}c_2(\|\mathbf{w}_{i+}\|^2 + b_{i+}^2) - \boldsymbol{\alpha}^T((\mathbf{B}\mathbf{w}_{i+} + \mathbf{e}_B b_{i+}) + \boldsymbol{\xi} - \mathbf{e}_B) - \boldsymbol{\beta}^T\boldsymbol{\xi}. \tag{11}$$

From the (Theorem 2.1), the KKT system of (9) is as follows:

$$\mathbf{A}_i^T(\mathbf{A}_i\mathbf{w}_{i+} + \mathbf{e}_{Ai}b_{i+}) + c_2\mathbf{w}_{i+} - \mathbf{B}^T\boldsymbol{\alpha} = \mathbf{0}, \tag{12}$$

$$\mathbf{e}_{Ai}^T(\mathbf{A}_i\mathbf{w}_{i+} + \mathbf{e}_{Ai}b_{i+}) + c_2 b_{i+} - \mathbf{e}_B^T\boldsymbol{\alpha} = 0, \tag{13}$$

$$c_1\mathbf{e}_B - \boldsymbol{\alpha} - \boldsymbol{\beta} = \mathbf{0}, \tag{14}$$

$$\boldsymbol{\alpha}^T((\mathbf{B}\mathbf{w}_{i+} + \mathbf{e}_B b_{i+}) + \boldsymbol{\xi} - \mathbf{e}_B) = \mathbf{0},\ \boldsymbol{\beta}^T\boldsymbol{\xi} = 0. \tag{15}$$

Defining $\mathbf{H}_i = [\mathbf{A}_i, \mathbf{e}_{Ai}]$, $\mathbf{G} = [\mathbf{B}, \mathbf{e}_B]$, $\mathbf{z}_{i+}^T = [\mathbf{w}_{i+}^T, b_{i+}]$, $i = 1, \ldots, k$, and $\mathbf{I}$ is the identity matrix of order $(n+1)$, from (12) and (13) we have

$$\mathbf{H}_i^T\mathbf{H}_i\mathbf{z}_{i+} + c_2\mathbf{I}\mathbf{z}_{i+} - \mathbf{G}^T\boldsymbol{\alpha} = \mathbf{0} \tag{16}$$

$$\Rightarrow \left[\mathbf{H}_i^T\mathbf{H}_i + c_2\mathbf{I}\right]\mathbf{z}_{i+} = \mathbf{G}^T\boldsymbol{\alpha} \tag{17}$$

$$\Rightarrow \mathbf{z}_{i+} = \left[\mathbf{H}_i^T\mathbf{H}_i + c_2\mathbf{I}\right]^{-1}\mathbf{G}^T\boldsymbol{\alpha}. \tag{18}$$

Substituting (18) into the Lagrangian (11), and combined with the conditions (14), and (15) we have the dual problem of (9) as follows:

$$\begin{cases} \max\limits_{\boldsymbol{\alpha}} & \mathbf{e}_B^T \boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T \mathbf{G}[\mathbf{H}_i^T \mathbf{H}_i + c_2 \mathbf{I}]^{-1} \mathbf{G}^T \boldsymbol{\alpha}, \\ \text{s.t.} & \mathbf{0} \le \boldsymbol{\alpha} \le c_1 \mathbf{e}_B. \end{cases} \tag{19}$$

In a similar way, by defining $\mathbf{G}_j = [\mathbf{B}_j, \mathbf{e}_{Bj}]$, $\mathbf{H} = [\mathbf{A}, \mathbf{e}_A]$, $\mathbf{z}_{j-}^T = [\mathbf{w}_{j-}^T, b_{j-}]$, $j = 1, \ldots, l$, we obtain the solutions of problem (10):

$$\mathbf{z}_{j-} = \left[\mathbf{G}_j^T \mathbf{G}_j + c_4 \mathbf{I}\right]^{-1} \mathbf{H}^T \boldsymbol{\gamma}, \tag{20}$$

where $\boldsymbol{\gamma}$ is the solution of the dual problem of (10) as follows:

$$\begin{cases} \max\limits_{\boldsymbol{\gamma}} & \mathbf{e}_A^T \boldsymbol{\gamma} - \frac{1}{2}\boldsymbol{\gamma}^T \mathbf{H}[\mathbf{G}_j^T \mathbf{G}_j + c_4 \mathbf{I}]^{-1} \mathbf{H}^T \boldsymbol{\gamma}, \\ \text{s.t.} & \mathbf{0} \le \boldsymbol{\gamma} \le c_3 \mathbf{e}_A. \end{cases} \tag{21}$$

3.2. *The nonlinear case*

The linear TSVM-MCD can also easily be extended when two classes are nonlinearly separable. Let $\Phi : \mathbb{R}^n \to \mathbb{H}$ be a nonlinear mapping, where $\mathbb{H}$ is a Hilbert space whose dimension is not less than n (maybe infinite-dimensional). Since $\mathbb{S} = \text{span}(\Phi(\mathbf{C}^T))$ is a subspace of $\mathbb{H}$ whose dimension does not exceed m, we can consider $\mathbb{S}$ as an Euclidean space and $\Phi : \mathbb{R}^n \to \mathbb{S}$. Suppose that after the clustering step on space $\mathbb{S}$, we obtain k clusters $\Phi(\mathbf{A}_1), \ldots, \Phi(\mathbf{A}_k)$ in the class $\Phi(\mathbf{A})$, each cluster $\Phi(\mathbf{A}_i)$ consists of m_{Ai} data points; and l clusters $\Phi(\mathbf{B}_1), \ldots, \Phi(\mathbf{B}_l)$ in the class $\Phi(\mathbf{B})$, each cluster $\Phi(\mathbf{B}_j)$ consists of m_{Bj} data points. In space $\mathbb{S}$, a hyperplane $\Phi(\mathbf{x}^T)\mathbf{h} + b = 0$ (with $\mathbf{h} \in \mathbb{S}$ being the normal vector) can be rewritten as $\Phi(\mathbf{x}^T)\Phi(\mathbf{C}^T)\mathbf{u} + b = 0$ for some vector $\mathbf{u} \in \mathbb{R}^m$. Therefore, by defining $\Phi(\mathbf{x}^T)\Phi(\mathbf{C}^T) = K(\mathbf{x}^T, \mathbf{C}^T)$, the hyperplane has the form $K(\mathbf{x}^T, \mathbf{C}^T)\mathbf{u} + b = 0$, K is a predefined kernel.[17]

TSVM-MCD determines k hyperplanes such that the $i-$th one: $K(\mathbf{x}^T,\ \mathbf{C}^T)\mathbf{u}_{i+} + b_{i+} = 0$ is closer to cluster $\Phi(\mathbf{A}_i)$ and far away from class $\Phi(\mathbf{B})$. It also determines l hyperplanes such that the $j-$th one: $K(\mathbf{x}^T,\ \mathbf{C}^T)\mathbf{u}_{j-} + b_{j-} = 0$ is closer to cluster $\Phi(\mathbf{B}_j)$ and far away from class $\Phi(\mathbf{A})$. Specifically, we have $(k + l)$ QPPs as follows:

$$\begin{cases} \min\limits_{\mathbf{u}_{i+}, b_{i+}, \boldsymbol{\xi}} & \frac{1}{2}\|K(\mathbf{A}_i, \mathbf{C}^T)\mathbf{u}_{i+} + \mathbf{e}_{Ai} b_{i+}\|^2 + \frac{c_1}{2}\mathbf{e}_B^T \boldsymbol{\xi} + \frac{c_2}{2}(\|\mathbf{u}_{i+}\|^2 + b_{i+}^2), \\ \text{s.t.} & (K(\mathbf{B}, \mathbf{C}^T)\mathbf{u}_{i+} + \mathbf{e}_B b_{i+}) + \boldsymbol{\xi} \ge \mathbf{e}_B, \boldsymbol{\xi} \ge \mathbf{0}; \end{cases} \tag{22}$$

$\mathbf{u}_{i+} \in \mathbb{R}^m$, $i = 1, \ldots, k$ and

$$\begin{cases} \min\limits_{\mathbf{u}_{j-}, b_{j-}, \boldsymbol{\eta}} & \frac{1}{2}\|K(\mathbf{B}_j, \mathbf{C}^T)\mathbf{u}_{j-} + \mathbf{e}_{Bj} b_{j-}\|^2 + \frac{c_3}{2}\mathbf{e}_A^T \boldsymbol{\eta} + \frac{c_4}{2}(\|\mathbf{u}_{j-}\|^2 + b_{j-}^2), \\ \text{s.t.} & (K(\mathbf{A}, \mathbf{C}^T)\mathbf{u}_{j-} + \mathbf{e}_A b_{j-}) + \boldsymbol{\eta} \ge \mathbf{e}_A, \boldsymbol{\eta} \ge \mathbf{0}; \end{cases} \tag{23}$$

$\mathbf{u}_{j-} \in \mathbb{R}^m$, $j = 1, \ldots, l$.

In an exactly similar way such as the linear case, two dual problems of (22) and (23) are as follows:

$$\begin{cases} \max\limits_{\boldsymbol{\alpha}} & \mathbf{e}_B^T\boldsymbol{\alpha} - \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{G}[\mathbf{H}_i^T\mathbf{H}_i + c_2\mathbf{I}]^{-1}\mathbf{G}^T\boldsymbol{\alpha}, \\ \text{s.t.} & \mathbf{0} \le \boldsymbol{\alpha} \le c_1\mathbf{e}_B, \end{cases} \tag{24}$$

and

$$\begin{cases} \max\limits_{\boldsymbol{\gamma}} & \mathbf{e}_A^T\boldsymbol{\gamma} - \frac{1}{2}\boldsymbol{\gamma}^T\mathbf{H}[\mathbf{G}_j^T\mathbf{G}_j + c_4\mathbf{I}]^{-1}\mathbf{H}^T\boldsymbol{\gamma}, \\ \text{s.t.} & \mathbf{0} \le \boldsymbol{\gamma} \le c_3\mathbf{e}_A. \end{cases} \tag{25}$$

Where, $\mathbf{H}_i = [K(\mathbf{A}_i, \mathbf{C}^T), \mathbf{e}_{Ai}]$, $\mathbf{G} = [K(\mathbf{B}, \mathbf{C}^T), \mathbf{e}_B]$, $\mathbf{G}_j = [K(\mathbf{B}_j, \mathbf{C}^T), \mathbf{e}_{Bj}]$, $\mathbf{H} = [K(\mathbf{A}, \mathbf{C}^T), \mathbf{e}_A]$, and we have solutions of (22), (23) as follows:

$$\mathbf{z}_{i+} = \left[\mathbf{H}_i^T\mathbf{H}_i + c_2\mathbf{I}\right]^{-1}\mathbf{G}^T\boldsymbol{\alpha}, i = \overline{1,k}, \tag{26}$$

$$\mathbf{z}_{j-} = \left[\mathbf{G}_j^T\mathbf{G}_j + c_4\mathbf{I}\right]^{-1}\mathbf{H}^T\boldsymbol{\gamma}, j = \overline{1,l}, \tag{27}$$

here, $\mathbf{z}_{i+}^T = [\mathbf{u}_{i+}^T, b_{i+}]$ $\mathbf{z}_{j-}^T = [\mathbf{u}_{j-}^T, b_{j-}]$, and $\mathbf{I}$ is the identity matrix of order $(m+1)$.

The classification function is now selected as:

$$f(\mathbf{x}) = \underset{+,\ -}{\text{argmin}}(f_+(\mathbf{x}),\ f_-(\mathbf{x})), \tag{28}$$

with

$$\begin{cases} f_+(\mathbf{x}) = \sum\limits_{i=1}^{k} \dfrac{m_{Ai}}{m_A}(K(\mathbf{x}^T, \mathbf{C}^T)\mathbf{u}_{i+} + b_{i+}); \\ f_-(\mathbf{x}) = \sum\limits_{j=1}^{l} \dfrac{m_{Bj}}{m_B}(K(\mathbf{x}^T, \mathbf{C}^T)\mathbf{u}_{j-} + b_{j-}). \end{cases} \tag{29}$$

Discussions

The main ideals: The SVM[1] seeks **a** hyperplane separating two classes (**A** and **B**) with largest margin. TSVM[11] and S-TSVM[12] seek **two** hyperplanes, each hyperplane is closer to one class, and far away from another class. TSVM-MCD seeks $(\mathbf{k} + \mathbf{l})$ hyperplanes (corresponding with the number of clusters in two classes) such that each hyperplane is closer to one cluster of one class, and far away from remaining class.

The methods: The SVM[1] seeks **a** hyperplane by solving the big QPP (see equation (1)). TSVM[11] and S-TSVM[12] seek **two** hyperplanes by solving two smaller QPPs (see equations (2), (3), (5), (6)). TSVM-MCD seeks $(k + l)$ hyperplanes by solving $(k + l)$ small QPPs (see equations (9), (10)).

The weights: The classifier of TSVM-MCD based on the weight that is proportional to the number of data points in each cluster (see equations (7), (8), (29)). While, the SVM, TSVM, S-TSVM don't care about the number of data points of each cluster in two classes.

The training times (or computational complexity): The constrains of big QPP of SVM[1] are all data points of two classes (m constrains). The constrains of each smaller QPP of TSVM,[11] S-TSVM[12] and TSVM-MCD are data points of one class (m_A or m_B constrains). That is why the training time of TSVM is four times faster than that of SVM (see TSVM[11]). The training time of S-TSVM and TSVM-MCD are slower than that of TSVM because S-TSVM and TSVM-MCD have to do clustering work in each class (**A** and **B**) before solving the QPPs. In the objective function, the S-TSVM[12] (see equations (5), (6)) must calculate the covariance matrix of all clusters (structural information with cluster granularity), that is why the training time of S-TSVM is slower than that of TSVM-MCD.

TSVM is the special case of TSVM-MCD: The TSVM-MCD includes two steps: the first step is to group data in each class, if both two class **A** and **B** are constructed by only one cluster, and let hyperparameters c_2, c_4 in TSVM-MCD equal to zero, then TSVM-MCD is exactly the TSVM[11] (see equations (2), (3), (9), (10)).

4. Experiments

In this section, we compare the training time, tested accuracy, and 10-fold cross-validated accuracy (10-fold CV) of TSVM-MCD against S-TSVM[12] and TSVM[11] on various UCI data sets.[18] All algorithms are settled by Python programming language and run on a desktop with an Intel Xeon E5, and 32GB RAM. All settings are uploaded to

- https://github.com/makeho8/Algorithms.

We implement these algorithms on UCI data sets[18] which have been experimented in.[11,12] We randomly selected 70% of each extracted data set for training and 30% for testing. We both used 10-fold cross-validation (CV) on the training set and tested accuracy on the testing set to evaluate the performance of all algorithms. All hyper-parameters such as c_1, c_2 of TSVM, $c_1, c_2, c_3, c_4, c_5, c_6$ of S-TSVM, c_1, c_2, c_3, c_4 of TSVM-MCD, and γ of RBF kernel are set to 1 to balance the roles of components in the QPP's objective functions of all algorithms.

Table 1 compares tested accuracy, and 10-fold CV accuracy with linear kernel between TSVM-MCD, S-TSVM, and TSVM on 9 small UCI data sets. Because the data sets are small, the training time of all algorithms is not much different, so we don't show the time in Table 1. From Table 1 we can see that the generalization performance of TSVM-MCD is better than that of TSVM and S-TSVM, it is expressed in tested accuracy. The tested accuracy of TSVM-MCD is higher than that of other algorithms in multi-clusters data cases. On the other cases, when each class is made up by one cluster (Ionosphere, Auto-mpg, Heart-disease, Credit-approval datasets), exploiting information about the structure with cluster granularity, and the number of data points in each cluster of TSVM-MCD are no longer makes significance. That is why the classification accuracy of all algorithms are similar.

The 10-fold CV accuracy of the algorithms is comparable and quite close to the tested accuracy. This shows that the TSVM-MCD algorithm is quite stable, not falling into the case of over-fitting and under-fitting with training data.

Table 1. Training time (s), tested accuracy (%), and 10-fold cross-validated accuracy (%) with a linear Kernel of TSVM, S-TSVM, and TSVM-MCD on UCI data sets.

Data sets	Algorithms		
	TSVM	S-TSVM	TSVM-MCD
$(m \times n)$ $(k \times l)$	Test(%) 10-fold CV (%)	Test(%) 10-fold CV (%)	Test(%) 10-fold CV (%)
Hepatitis (155×19) (2×1)	87.2 81.7 +/- 12.1	87.2 81.7 +/- 12.1	**89.4** 82.5 +/- 11.0
Liver-disorders (345×5) (1×2)	77.9 73.4 +/- 8.2	77.9 73.4 +/- 8.2	**79.8** 75.9 +/- 8.5
Ionosphere (351×34) (1×1)	88.7 86.9 +/- 7.9	88.7 87.4 +/- 7.6	88.7 87.4 +/- 7.6
Glioma-grading (839×25) (2×2)	85.3 87.7 +/- 4.8	85.3 87.7 +/- 4.8	**86.1** 87.1 +/- 4.6
Auto-mpg (398×7) (1×1)	90.0 90.7 +/- 4.8	90.0 90.7 +/- 4.8	90.0 89.9 +/- 4.4
Automobile (205×25) (1×2)	90.3 84.6 +/- 10.0	90.3 84.6 +/- 10.0	**93.5** 80.4 +/- 11.0
Heart-disease (303×13) (1×1)	81.3 82.9 +/- 7.2	81.3 82.9 +/- 7.2	81.3 80.1 +/- 7.7
Heart-failure (299×12) (2×1)	78.9 84.7 +/- 9.9	78.9 84.7 +/- 9.9	77.8 82.8 +/- 10.9
Credit-approval (690×46) (1×1)	84.1 86.3 +/- 2.8	84.1 86.3 +/- 2.8	84.1 84.1 +/- 6.1

Where m is the data point number of each data set, n is the dimension of the data, k is the number of clusters in the class $\mathbf{A}$, and l is the number of clusters in class $\mathbf{B}$, respectively.

Table 2 compares the training time, tested accuracy, and 10-fold CV accuracy with RBF kernel between TSVM-MCD, S-TSVM, and TSVM on 6 small UCI data sets. From Table 2 we can see that the training time of TSVM-MCD and S-TSVM are slower than that of TSVM for the nonlinear case of all 6 data sets. This is because the S-TSVM and TSVM-MCD have to cluster the data and process the problem on each cluster. The training time of TSVM-MCD is faster than that of S-TSVM because the S-TSVM must calculate the covariance matrices of all clusters in two classes. That is the cost to obtain a more precise classification. We can see that the classification accuracy of TSVM-MCD, and S-TSVM are better than that of TSVM in most data sets.

Note that, the number of clusters ($k \times l$, k is the number of clusters in the class $\{+\}$ and l is in the class $\{-\}$) in each class (shows in Table 1 and Table 2) are auto-detected by using elbow method.

Table 2. Training time (s), tested accuracy (%), and 10-fold cross-validated accuracy (%) with an RBF Kernel of TSVM, S-TSVM, and TSVM-MCD on UCI data sets

Data sets	Algorithms		
	TSVM	S-TSVM	TSVM-MCD
($m \times n$) ($k \times l$)	Test(%) 10-fold CV (%) Time (s)	Test(%) 10-fold CV (%) Time (s)	Test(%) 10-fold CV (%) Time (s)
Liver-disorders	79.8	79.8	**81.7**
(345×5)	79.2 +/- 5.9	75.9 +/- 8.3	75.1 +/- 7.9
(1×2)	1.40	3.55	2.56
Ionosphere	86.8	91.5	91.5
(351×34)	84.8 +/- 9.2	95.5 +/- 3.4	95.5 +/- 4.2
(1×1)	1.27	3.29	1.80
Glioma-grading	77.0	83.3	**84.1**
(839×25)	70.9 +/- 5.8	85.5 +/- 5.3	85.5 +/- 6.8
(2×2)	7.58	20.02	16.29
Auto-mpg	91.7	91.7	91.7
(398×7)	91.0 +/- 4.3	90.7 +/- 3.6	89.2 +/- 4.8
(1×1)	1.69	4.50	2.39
Heart-disease	73.6	76.9	76.9
(303×13)	71.1 +/- 11.5	83.0 +/- 8.1	82.0 +/- 5.6
(1×1)	0.99	2.46	1.62
Credit-approval	69.6	85.0	85.0
(690×46)	68.8 +/- 5.0	85.7 +/- 4.0	81.8 +/- 4.5
(1×1)	4.82	12.95	8.69

5. Conclusion

This paper proposes a new Twin Support Vector Machine with Multi-Clusters Data (TSVM-MCD) for binary classification problems, using a cluster-vs-class strategy. TSVM-MCD has two steps: The first step is grouping in each class by Ward's linkage clustering method; The second is model learning. The classifier is based on weighted average distances from the data point to the cluster's representative hyperplanes. TSVM-MCD has a slower execution time than that of TSVM, but faster than S-TSVM in nonlinear cases. Regarding classification accuracy, TSVM-MCD is better than TSVM, and S-TSVM in most data sets for both linear and nonlinear cases. For binary classification problems with multi-clusters data, in which each class contains many clusters, and each cluster has individual distribution trends and data points, the TSVM-MCD algorithm is more effective in generalized performance. This new method may not be suitable for multi-classes problems. However, it seems useful in solving the classification problem with imbalanced data.

Acknowledgments The Authors would like to thank the referees for their helpful comments and valuable suggestions.

References

1. V. Vapnik, *The Natural Of Statistical Learning Theory* (Springer, Verlag New York, 1995).
2. G. Fung and O. L. Mangasarian, Proximal support vector machine, kdd '01: Proceedings of the 7-th acm sigkdd international conference on knowledge discovery and data mining (NY, United States, San Francisco California, 2001), pp. 77–86.
3. W. Noble, *Support Vector Machine Applications in Computational Biology* (MIT Press, Seatle, 2004).
4. M. Adancon and M. Cheriet, Model selection for the ls-svm. application to handwriting recognition, *Pattern Recognition* **42** (2009) 3264–3270.
5. Y. Tian, Y. Shi and Y. Liu, Recent advances on support vector machines research, *Technological and Economic Development of Economy* **18** (2012) 5–33.
6. D. Tomar and S. Agarwal, Twin support vector machine: A review from 2007 to 2014, *Egyptian Informatics Journal* **16** (2015) 55–69.
7. J. Cervantes, F. Lamont, L. Mazahua and A. Lopez, A comprehensive survey on support vector machine classification: Applications, challenges and trends, *Neurocomputing* **408** (2020) 189–215.
8. X. Pan, Y. Luo and Y. Xu, K-nearest neighbor based structural twin support vector machine, *Knowledge-Based Systems* **88** (2015) 34–44.
9. X. Xie and S. Sun, Multitask centroid twin support vector machines, *Neurocomputing* **149** (2015) 1085–1091.
10. B. Mei and Y. Xu, Multi-task least squares twin support vector machine for classification, *Neurocomputing* **338** (2019) 26–33.
11. Jayadeva, R. Khemchandani and S. Chandra, Twin support vector machines for pattern classification, *IEEE Transactions on Pattern Analysis and Machine intelligence* **29** (2007) 905–910.
12. Z. Qi, Y. Tian and Y. Shi, Structural twin support vector machine for classification, *Knowledge-Based Systems* **43** (2013) 74–81.

13. W. Sun and Y.-X. Yuan, *Optimization theory and methods: nonlinear programming* (Springer, New York, NY, 2006).
14. N. T. Cuong, Hierarchical multi twin support vector machine, *Hue University Journal of Science: Techniques and Technology* **130**(2B) (2021).
15. H. Xue, S. Chen and Q. Yang, Structural regularized support vector machine: A framework for structural large margin classifier, *IEEE Transactions on Neural Networks* **22** (2011) 573–587.
16. J. H. Ward, Hierarchical grouping to optimize an objective function, *Journal of the American Statistical Association* **58**(301) (1963) 236–244.
17. B. Scholkopf and A. J. Smola, *Learning with kernel* (MIT Press, Cambridge, Massachusetts, London, 2018).
18. UCI, Machine learning repository (2007), http://archive.ics.uci.edu/ml/machine-learning-databases/.

Vietnam Journal of Computer Science

Reply to reviewer

Paper title: Twin Support Vector Machine with Multi-Clusters Data

Ms. No. WSPC-VJCS-D-24-00193

Authors

We would like to thank the referees for their helpful comments and valuable suggestions. The paper has been revised according to their requirements (the changes are highlighted in the article). The detailed modifications are listed below.

- The last paragraph of the Introduction section should be relocated to the Background section for improved organization and clarity.

  R: We have relocated the last paragraph of the Introduction section as suggestion.

- In the abstract, the statement "Both theorized and experimentally, we show the comparison of TSVM-MCD with two improvements of SVM" lacks a theoretical comparison of the algorithms, as indicated.

- The abstract needs rephrasing to better highlight the originality and contribution of the manuscript beyond just comparing algorithms.

  R: They are true that, so we have rewritten the abstract and added a discussion section to take a more detailed look at the algorithms.

- Meaningful results should be thoroughly explained, particularly instances where the performance of algorithms is equal. Providing insights into why this occurred can enhance the manuscript's comprehensibility.

  R: We have added more explained for the performance of algorithms.

- The rationale behind selecting the two improved SVM algorithms should be elucidated. Additionally, clarification is needed regarding any significance associated with the proposals of Structural Twin Support Vector Machine in 2013 and Twin Support Vector Machines in 2007.

  R: We have more explained for these two algorithms.

- In the introduction section, the authors should mention the main contributions and effectiveness of the proposed method.

- In the introduction section, the authors should mention the main contributions and effectiveness of the proposed method.

  R: We have added these suggestions in the Introduction section.

- The experimental data is from only one source.The authors may consider adding more data sources to demonstrate the effectiveness of the algorithm. The authors consider further comparisons with other algorithms such as LS-TWSVM (the reviewer mean that Least Square Twin Support Vector Machine).

  R: We know more difference data source. However in this paper, we introduces a novel approach for addressing binary classification problems, and in the original results they used these data sets. We also know more new research that are related to TSVM and S-TSVM, but they not closed to the novel approach of TSVM and S-TSVM. For example, the LS-TSVM has the same approach, but it find the global solution by using "least square" method, while TSVM, S-TSVM and TSVM-MCD solving the Quadratic Programming Problem to find the optimal solution.

- Why did the author not compare with the author's previous proposed method HM-TWSVM (mean that Hierarchical Multi Twin Support Vector Machine).

  R: This is the first result when we study on SVMs. That is not a good ideal, because we thought that all clusters in each class have the same role in the classification problem. We have added this paper in the References section, and explained why in the Introduction.

Thank so much

Authors