

# Proximal Support Matrix Machine

Wan Zhang, Yulan Liu

School of Mathematics and Statistics, Guangdong University of Technology, Guangzhou, China

Email: wan\_zhang0315@163.com, ylliu@gdut.edu.cn

**How to cite this paper:** Zhang, W. and Liu, Y.L. (2022) Proximal Support Matrix Machine. *Journal of Applied Mathematics and Physics*, 10, 2268-2291.

<https://doi.org/10.4236/jamp.2022.107155>

**Received:** June 21, 2022

**Accepted:** July 25, 2022

**Published:** July 28, 2022

Copyright © 2022 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## Abstract

In this paper, we have proposed a novel model called proximal support matrix machine (PSMM), which is mainly based on the models of proximal support vector machine (PSVM) and low rank support matrix machine (LRSMM). In design, the PSMM model has comprehensively considered both the relationship between samples of the same class and the structure of rows or columns of matrix data. To a certain extent, our novel model can be regarded as a synthesis of the PSVM model and the LRSMM model. Since the PSMM model is an unconstrained convex problem in essence, we have established an alternating direction method of multipliers algorithm to deal with the proposed model. Finally, since a great deal of experiments on the minst digital database show that the PSMM classifier has a good ability to distinguish two digits with little difference, it encourages us to conduct more complex experiments on MIT face database, INRIA person database, the students face database and Japan female facial expression database. Meanwhile, the final experimental results show that PSMM performs better than PSVM, twin support vector machine, LRSMM and linear twin multiple rank support matrix machine in the demanding image classification tasks.

## Keywords

PSMM, PSVM, LRSMM, The Alternating Direction Method of Multipliers Algorithm, Image Classification Tasks

## 1. Introduction

Classification is an important research field of machine learning, in which the support vector classifier is a common and important tool. There are many excellent designs of support vector classifiers. For example, the proximal support vector machine (PSVM) [1] has considered the relationship between samples within a class, which makes congener samples have aggregation effect. Subsequently, a great deal of vector classifiers were developed on the basis of PSVM

classifier, see [2] [3] [4] [5] [6]. Moreover, deferent from PSVM with single model, the twin support vector machine (TSVM) [7] has designed two relatively small SVM-type models, which makes samples can be quickly classified by a pair of nonparallel separating planes. Inspired by TSVM, there also have developed a lot of twin-type vector classifiers, see [8] [9] [10] [11]. For more innovative designs of support vector classifiers, see [12] [13] [14] [15]. However, with the progress of modern science and technology, data to be processed is often expressed as matrices rather than vectors in application. When using vector classifiers on matrix data, the input matrices have to be reshaped into vectors, which will destroy the structure of rows or columns and thus lose important classification information unique to matrix data. Additionally, the reconstruction of matrix will inevitably lead to the superposition of rows or columns, resulting in a high-dimensional vector. It is not conducive to conduct large-scale testing in matrix classification tasks.

Realizing the above problems, many works have been proposed to extend the classification strategy of support vector machines to matrix space, which can get the eminent results in matrix classification problems. For example, L. Luo *et al.* have proposed low rank support matrix machine (LRSMM) [16] on the basis of the soft-margin support vector machine (Soft-SVM) [17], which has considered the correlation of rows and columns of matrix samples by using nuclear norm as the convex approximation of matrix rank. X. Gao *et al.* have proposed linear twin multiple rank support matrix machine (LTMRSMM) [18] on the basis of TSVM [7], which has considered the practical situation that matrix data are multiple rank. Q. Zheng *et al.* have proposed the robust support matrix machine [19] on the basis of the bilinear support vector machine [20], which has considered the spatial-temporal structural information of the input matrices and thus can eliminate non-standard noises of samples. For more cutting-edge works, see [21] [22] [23] [24] [25]. The designs of these SMM-type models provide us a new idea to solve matrix classification problems.

In this paper, to generalize the classification method of PSVM [1] to matrix space, we propose the proximal support matrix machine (PSMM) on the basis of PSVM [1] and LRSMM [16], and we give the novel model as follows:

$$\min_{(\mathbf{W}, b) \in \mathbb{R}^{l \times m} \times \mathbb{R}} \frac{1}{2} (\|\mathbf{W}\|_F^2 + b^2) + \tau \|\mathbf{W}\|_* + \frac{C}{2} \sum_{i=1}^n [1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b)]^2$$

where  $\mathbf{X}_i \in \mathbb{R}^{l \times m}$ ,  $y_i \in \mathbb{R}$  are given matrix sample and category label for each  $i$  and  $C, \tau$  are two given penalty parameter. In design, our model has absorbed the advantages of PSVM and LRSMM, which has comprehensively considered the relationship between samples within a class and the structure of rows or columns of matrix samples. Meanwhile, since the novel model is essentially a convex problem, it encourages us to establish an alternating direction method of multipliers (ADMM) algorithm [26] to deal with the PSMM model. Finally, we mainly focus on image classification, which is a very significant problem of matrix classification. We conduct a series of comparative experiments on minst

digital database [27], MIT face database, INRIA person database [28], the students face database [29] and Japan female facial expression database [30]. The final experimental results show that our method performs better than PSVM [1], TSVM [7], LRSMM [16] and LTMRSMM [18] in demanding image classification tasks.

The remainder of this paper is arranged as follows. In Section 2, we give the notations and lemma used in our paper. In Section 3, we recall the models of PSVM [1] and LRSMM [16] related to our novel model. In Section 4, the PSMM model is proposed and then an ADMM algorithm is established to solve it. In Section 5, some experimental analyses are presented to verify the performance of the PSMM classifier. In Section 6, our work is summarized.

## 2. Preliminaries

As a matter of convenience, we give the notations and lemma in this section, which will be used in this paper.

For a given matrix  $A \in \mathbb{R}^{l \times m}$  with  $\text{rank}(A) = r_A$  and the singular values  $\sigma_1(A) \geq \dots \geq \sigma_{r_A}(A) > 0$ , we define the singular value decomposition of the matrix  $A$  by

$$A = U_A \Sigma_A V_A^T \in \mathbb{R}^{l \times m}, \tag{1}$$

where  $U_A \in \mathbb{R}^{l \times r_A}$ ,  $V_A \in \mathbb{R}^{m \times r_A}$  and  $\Sigma_A$  is a diagonal matrix with diagonal entries  $\sigma_1(A), \dots, \sigma_{r_A}(A)$ . And, the Frobenius norm of the matrix  $A$  is denoted as

$$\|A\|_F = \left( \sum_{i,j} [A]_{ij}^2 \right)^{\frac{1}{2}} = \left( \sum_{i=1}^{r_A} \sigma_i(A)^2 \right)^{\frac{1}{2}}$$

and the nuclear norm of  $A$  is denoted as

$$\|A\|_* = \sum_{i=1}^{r_A} \sigma_i(A)$$

whose subdifferential is denoted by  $\partial \|A\|_*$ . We reshape  $A$  into a vector  $a \in \mathbb{R}^{lm}$  with a traditional and common matrix reconstruction approach:

$$a := \text{vec}(A) := ([A]_{11}, \dots, [A]_{ll}, [A]_{21}, \dots, [A]_{lm})^T$$

and define the Frobenius norm  $\|a\| = \left( \sum_{i,j} [A]_{ij}^2 \right)^{\frac{1}{2}} = \|A\|_F$ .

**Lemma 1.** (see [31], Theorem 2.1) Given a matrix  $A \in \mathbb{R}^{l \times m}$  with  $\text{rank}(A) = r_A$  and the singular value decomposition as defined in (1). For  $\gamma > 0$ , the following problem

$$\min_{Q \in \mathbb{R}^{l \times m}} \gamma \|Q\|_* + \frac{1}{2} \|Q - A\|_F^2$$

has unique closed-form solution, denoted by  $Q^*$ , as follows:

$$Q^* = \mathcal{D}_\gamma(A) := U_A D_\gamma V_A^T$$

where  $D_\gamma$  is a diagonal matrix with  $[\sigma_1(\mathbf{A}) - \gamma]_+, \dots, [\sigma_{r_A}(\mathbf{A}) - \gamma]_+$  as diagonal entries and  $[t]_+ = \max\{t, 0\}$  for any  $t \in \mathbb{R}$ .

In this paper, we give a matrix data set  $T_m = \{(\mathbf{X}_i, y_i)\}_{i=1}^n$  with  $\mathbf{X}_i \in \mathbb{R}^{l \times m}$  and  $y_i \in \{-1, 1\}$ , and denote the label vector as  $\mathbf{y} = (y_1, \dots, y_n)^T \in \mathbb{R}^n$ . For the convenience of calculation, we define a linear mapping  $\mathcal{A}: \mathbb{R}^{l \times m} \rightarrow \mathbb{R}^n$  by

$$\mathcal{A}(\mathbf{A}) = (\langle \mathbf{A}, \mathbf{X}_1 \rangle, \dots, \langle \mathbf{A}, \mathbf{X}_n \rangle)^T \in \mathbb{R}^n$$

and denote  $\circ$  as the Hadamard product between two matrices. Additionally, we also denote the  $n$ -order identity matrix as  $\mathbf{I}$  and define an all one vector  $\mathbf{e} = (1, \dots, 1)^T \in \mathbb{R}^n$ .

### 3. Related Work

The design inspiration of PSMM proposed in this paper mainly comes from PSVM [1] and LRSMM [16]. In order to show the rationality and advantages of the PSMM model in design, we review and analyze the models of PSVM [1] and LRSMM [16] in this section.

#### 3.1. Proximal Support Vector Machine (PSVM)

Different from Soft-SVM model [17], the PSVM model is described in  $(\mathbf{w}, b)$  space of  $\mathbb{R}^{lm} \times \mathbb{R}$ . Additionally, PSVM defines a pair of proximal hyperplanes  $\langle \mathbf{w}, \mathbf{x} \rangle + b = \pm 1$  and constrains samples of the same class to be as close to the same proximal plane as possible. This is because the higher the fitting degree of proximal planes to the samples, the more obvious the boundary between the two types of samples, and then the better the classification effect of classification hyperplane. The strategy of PSVM can be presented as the following model, for details, see [1]:

$$\min_{(\mathbf{w}, b)} \frac{1}{2} (\|\mathbf{w}\|^2 + b^2) + \frac{c}{2} \sum_{i=1}^n [1 - y_i (\langle \mathbf{w}, \mathbf{x}_i \rangle + b)]^2 \quad (2)$$

where  $\mathbf{x}_i := \text{vec}(\mathbf{X}_i) \in \mathbb{R}^{lm}$  for each  $i$  is the reconstructed vector sample,  $(\mathbf{w}, b) \in \mathbb{R}^{lm} \times \mathbb{R}$  is the decision variable and  $c \in \mathbb{R}$  is a given penalty parameter. A new input  $\mathbf{X} \in \mathbb{R}^{l \times m}$  has to be reshaped into a vector  $\mathbf{x} \in \mathbb{R}^{lm}$ , and then it can be assigned by the decision function:  $y_v(\mathbf{x}) = \text{sign}(\langle \mathbf{w}, \mathbf{x} \rangle + b)$ , where  $\text{sign}(\cdot)$  is the symbolic function.

Compared with Soft-SVM [17], PSVM [1] has considered the relationship between samples within a class, that samples of the same class are required to be close together, so that congener samples have aggregation effect. The advantage of this consideration is that the distribution trend of congener samples can be grasped, which makes PSVM have a good performance in prediction. Meanwhile, the PSVM model (2) is an unconstrained quadratic programming problem in essence, so that the cost of PSVM classifier is lower than that of Soft-SVM classifier. Thus, PSVM has more advantage than Soft-SVM in large database classification, for details, see [1].

### 3.2. Low Rank Support Matrix Machine (LRSMM)

Different from vector data, each matrix data has an identifying intrinsic structure, which reveals significant category information in matrix classification. As an important structural information, the correlation of rows or columns of matrix is closely related to the rank of the regression matrix  $\mathbf{W}$ . Meanwhile, it is an effective measure for filtering the redundant information of samples to impose the low rank constraint on  $\mathbf{W}$  in application. However, matrix rank minimization problem is a typical NP-hard problem. Realizing this difficulty, inspired by the idea that the nuclear norm is the best convex approximation of matrix rank, LRSMM was developed on the basis of Soft-SVM [17], which has achieved the low-rank target of samples by minimizing the nuclear norm of  $\mathbf{W}$ . The LRSMM model is presented as follows, for details, see [16]:

$$\min_{\mathbf{W}, b} \frac{1}{2} \|\mathbf{W}\|_F^2 + \tau \|\mathbf{W}\|_* + C \sum_{i=1}^n [1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b)]_+ \quad (3)$$

where  $\mathbf{W} \in \mathbb{R}^{l \times m}$ ,  $b \in \mathbb{R}$  are decision variables,  $C \in \mathbb{R}$  and  $\tau \in \mathbb{R}_{++}$  are two given penalty parameters. A new input  $\mathbf{X} \in \mathbb{R}^{l \times m}$  can be assigned by the decision function:  $y_m(\mathbf{X}) = \text{sign}(\langle \mathbf{W}, \mathbf{X} \rangle + b)$ .

### 4. Proximal Support Matrix Machine (PSMM)

In this section, we propose a novel matrix classification method PSMM on the basis of PSVM [1] and LRSMM [16], which has generalized the classification method of PSVM to matrix space and also has inherited the design in LRSMM of constraining matrix samples to be low-rank. The formulation of our novel model is given as follows:

$$\min_{(\mathbf{W}, b)} \frac{1}{2} (\|\mathbf{W}\|_F^2 + b^2) + \tau \|\mathbf{W}\|_* + \frac{C}{2} \sum_{i=1}^n [1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b)]^2 \quad (4)$$

where  $(\mathbf{W}, b) \in \mathbb{R}^{l \times m} \times \mathbb{R}$  is the decision variable,  $C \in \mathbb{R}$  and  $\tau \in \mathbb{R}_{++}$  are two given penalty parameters. In order to take into consideration the relationship between congener samples, PSMM also defines a pair of proximal hyperplanes  $\langle \mathbf{W}, \mathbf{X} \rangle + b = \pm 1$ . And, the formula  $[1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b)]^2$  is used to calculate the Euclidean distance between the  $i$ th sample and the proximal hyperplane. In (4), we can achieve the target that samples of the same class should gather near the same proximal plane by minimizing the sum of these Euclidean distances. Additionally, to meet the need that matrix samples should be low-rank in application, the nuclear norm of the regression matrix  $\mathbf{W}$  is introduced into our objective function in the inspiration of LRSMM [16].

The design of the PSMM model (4) has many advantages. On the one hand, since the nuclear norm of  $\mathbf{W}$  is the only nonsmooth part in the PSMM model, PSMM algorithm presented below will be simpler than LRSMM algorithm [16] in framework design. And, in the following experiments, it can be found that the cost of PSMM classifier is much lower than that of LRSMM classifier. On the other hand, compared with PSVM [1], PSMM has taken into account the intrinsic

sis structure of matrix data, so that the performance of PSMM classifier is better than that of PSVM classifier in the following matrix classification tasks. Thus, to a certain degree, the novel model (4) is a synthesis of PSVM model (2) and LRSMM model (3).

#### 4.1. Model Solution

Since the PSMM model (4) is a convex problem, we can devise an ADMM-type algorithm [26] to solve the proposed novel model. Firstly, the PSMM model (4) can be equivalently written in the following form:

$$\begin{aligned} \min_{(\mathbf{W}, b), \mathbf{S}} \quad & \frac{1}{2} (\|\mathbf{W}\|_F^2 + b^2) + \tau \|\mathbf{S}\|_* + \frac{C}{2} \sum_{i=1}^n [1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b)]^2 \\ \text{s.t.} \quad & \mathbf{W} = \mathbf{S}. \end{aligned} \quad (5)$$

And, its corresponding augmented Lagrangian function is given as follows:

$$\begin{aligned} L_\rho((\mathbf{W}, b), \mathbf{S}; \mathbf{\Lambda}) = & \frac{1}{2} (\|\mathbf{W}\|_F^2 + b^2) + \tau \|\mathbf{S}\|_* + \frac{C}{2} \sum_{i=1}^n [1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b)]^2 \\ & + \langle \mathbf{\Lambda}, \mathbf{S} - \mathbf{W} \rangle + \frac{\rho}{2} \|\mathbf{S} - \mathbf{W}\|_F^2 \end{aligned} \quad (6)$$

where  $\mathbf{\Lambda} \in \mathbb{R}^{l \times m}$  is the Lagrangian multiplier of the problem (5) and  $\rho > 0$  is a given parameter of the penalty term. Consequently, we can take advantage of the augmented Lagrangian function (6) to establish the iterative framework of PSMM algorithm, which is based on the framework of ADMM. Given the state variables  $((\mathbf{W}^k, b^k), \mathbf{S}^k; \mathbf{\Lambda}^k)$  at iteration  $k$ , the proposed framework for updating variable status is given as follows:

$$\begin{aligned} (\mathbf{W}^{k+1}, b^{k+1}) = \arg \min_{\mathbf{W}, b} \left\{ \frac{1}{2} (\|\mathbf{W}\|_F^2 + b^2) + \frac{C}{2} \sum_{i=1}^n [1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b)]^2 \right. \\ \left. + \langle \mathbf{\Lambda}^k, \mathbf{S}^k - \mathbf{W} \rangle + \frac{\rho}{2} \|\mathbf{S}^k - \mathbf{W}\|_F^2 \right\} \end{aligned} \quad (7)$$

$$\mathbf{S}^{k+1} = \arg \min_{\mathbf{S}} \left\{ \tau \|\mathbf{S}\|_* + \langle \mathbf{\Lambda}^k, \mathbf{S} - \mathbf{W}^{k+1} \rangle + \frac{\rho}{2} \|\mathbf{S} - \mathbf{W}^{k+1}\|_F^2 \right\} \quad (8)$$

$$\mathbf{\Lambda}^{k+1} = \mathbf{\Lambda}^k + \rho (\mathbf{S}^{k+1} - \mathbf{W}^{k+1}) \quad (9)$$

Note that we have to solve the two optimization problems (7) and (8) on each update. Thus, it is necessary to determine their explicit solutions, which can reduce the cost of our algorithm to a great extent.

Solving (7) can be transformed into solving the following model:

$$\begin{aligned} \min_{\mathbf{W}, b, \mathbf{x}} \quad & \frac{1}{2} (\|\mathbf{W}\|_F^2 + b^2) + \frac{C}{2} \|\mathbf{x}\|^2 - \langle \mathbf{\Lambda}^k, \mathbf{W} \rangle + \frac{\rho}{2} \|\mathbf{S}^k - \mathbf{W}\|_F^2 \\ \text{s.t.} \quad & 1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b) = \xi_i, \quad i = 1, \dots, n. \end{aligned} \quad (10)$$

That is because the optimal partial variable of (10), uniformly denoted by  $(\mathbf{W}^{k+1}, b^{k+1})$ , is actually the optimal solution of the problem (7). Since the model (10) is essentially a smooth convex optimization problem only with equality constraints, we can solve this problem by its dual problem. The Lagrange func-

tion of (10) is given as follows:

$$L(\mathbf{W}, b, \boldsymbol{\xi}; \boldsymbol{\lambda}) = \frac{1}{2} (\|\mathbf{W}\|_F^2 + b^2) + \frac{C}{2} \|\boldsymbol{\xi}\|^2 - \langle \boldsymbol{\Lambda}^k, \mathbf{W} \rangle + \frac{\rho}{2} \|\mathbf{S}^k - \mathbf{W}\|_F^2 + \sum_{i=1}^n \lambda_i [1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b) - \xi_i]$$

where  $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_n)^T \in \mathbb{R}^n$  is the Lagrange multiplier of (10). For each  $k$ , we define the following matrices:

$$\tilde{\mathbf{Y}} = \mathbf{y}\mathbf{y}^T, \quad \tilde{\mathbf{X}} = \tilde{\mathbf{Y}} \circ [\mathcal{A}(\mathbf{X}_1), \dots, \mathcal{A}(\mathbf{X}_n)]$$

$$\tilde{\mathbf{S}}^k = \mathbf{y} \circ \mathcal{A}(\mathbf{S}^k), \quad \tilde{\boldsymbol{\Lambda}}^k = \mathbf{y} \circ \mathcal{A}(\boldsymbol{\Lambda}^k)$$

Consequently, the Karush-Kuhn-Tucker (KKT) conditions for the model (10) are given as follows:

$$\begin{cases} \mathbf{W} = \frac{1}{1+\rho} \left( \rho \mathbf{S}^k + \boldsymbol{\Lambda}^k + \sum_{i=1}^n y_i \lambda_i \mathbf{X}_i \right) \\ b = \sum_{i=1}^n y_i \lambda_i, \xi_i = \frac{1}{C} \lambda_i, i = 1, \dots, n \\ 1 - y_i (\langle \mathbf{W}, \mathbf{X}_i \rangle + b) - \xi_i = 0, i = 1, \dots, n \end{cases} \quad (11)$$

According to duality theory and (11), after simple calculation, we can get the dual problem of (10) as follows:

$$\max_{\boldsymbol{\lambda} \in \mathbb{R}^n} d(\boldsymbol{\lambda}) \quad (12)$$

where  $d(\boldsymbol{\lambda})$  has the following expression:

$$d(\boldsymbol{\lambda}) = \langle \boldsymbol{\lambda}, \mathbf{e} \rangle - \frac{\rho}{1+\rho} \langle \boldsymbol{\lambda}, \tilde{\mathbf{S}}^k \rangle - \frac{1}{1+\rho} \langle \boldsymbol{\lambda}, \tilde{\boldsymbol{\Lambda}}^k \rangle - \frac{1}{C} \|\boldsymbol{\lambda}\|^2 - \frac{1}{2} \langle \boldsymbol{\lambda}, \tilde{\mathbf{Y}} \boldsymbol{\lambda} \rangle - \frac{1}{2(1+\rho)} \langle \boldsymbol{\lambda}, \tilde{\mathbf{X}} \boldsymbol{\lambda} \rangle$$

and the optimal solution of (12) is denoted by  $\boldsymbol{\lambda}^k$ . Since the dual problem (12) is essentially an unconstrained quadratic programming problem, by the optimality theorem, we find that solving (12) is equivalent to solve the following system:

$$\nabla_{\boldsymbol{\lambda}} d = \mathbf{e} - \frac{\rho}{1+\rho} \tilde{\mathbf{S}}^k - \frac{1}{1+\rho} \tilde{\boldsymbol{\Lambda}}^k - \frac{2}{C} \boldsymbol{\lambda} - \tilde{\mathbf{Y}} \boldsymbol{\lambda} - \frac{1}{1+\rho} \tilde{\mathbf{X}} \boldsymbol{\lambda} = \mathbf{0},$$

i.e.,

$$\left( \frac{2}{C} \mathbf{I} + \tilde{\mathbf{Y}} + \frac{1}{1+\rho} \tilde{\mathbf{X}} \right) \boldsymbol{\lambda} = \mathbf{e} - \frac{\rho}{1+\rho} \tilde{\mathbf{S}}^k - \frac{1}{1+\rho} \tilde{\boldsymbol{\Lambda}}^k. \quad (13)$$

If the coefficient matrix of (13) is invertible, then the optimal solution  $\boldsymbol{\lambda}^k$  of (12) has the explicit exact expression as follows:

$$\boldsymbol{\lambda}^k = \left( \frac{2}{C} \mathbf{I} + \tilde{\mathbf{Y}} + \frac{1}{1+\rho} \tilde{\mathbf{X}} \right)^{-1} \left( \mathbf{e} - \frac{\rho}{1+\rho} \tilde{\mathbf{S}}^k - \frac{1}{1+\rho} \tilde{\boldsymbol{\Lambda}}^k \right).$$

Therefore, by the KKT system (11), the optimal partial variable of (10)

$(\mathbf{W}^{k+1}, b^{k+1})$  has the following explicit exact expression:

$$\begin{cases} \mathbf{W}^{k+1} = \frac{1}{1+\rho} \left( \rho \mathbf{S}^k + \mathbf{\Lambda}^k + \sum_{i=1}^n y_i \lambda_i^k \mathbf{X}_i \right) \\ b^{k+1} = \sum_{i=1}^n y_i \lambda_i^k \end{cases}$$

which means that the iterative format (7) can be written in a more concise form:

$$(\mathbf{W}^{k+1}, b^{k+1}) = \left( \frac{1}{1+\rho} \left( \rho \mathbf{S}^k + \mathbf{\Lambda}^k + \sum_{i=1}^n y_i \lambda_i^k \mathbf{X}_i \right), \sum_{i=1}^n y_i \lambda_i^k \right).$$

Additionally, the problem (8) can be equivalently written as follows:

$$\arg \min_s \left\{ \frac{\tau}{\rho} \|\mathbf{S}\|_* + \frac{1}{2} \left\| \mathbf{S} - \left( \mathbf{W}^{k+1} - \frac{1}{\rho} \mathbf{\Lambda}^k \right) \right\|_F^2 \right\}. \tag{14}$$

By Lemma 1, we can deduce the closed-form solution of (14), so that the iterative format (8) has more straightforward expression as follows:

$$\mathbf{S}^{k+1} = \mathcal{D}_{\frac{\tau}{\rho}} \left( \mathbf{W}^{k+1} - \frac{1}{\rho} \mathbf{\Lambda}^k \right).$$

### 4.2. Convergence

Since the model (5) is a convex problem only with equality constraints, then the convergence property of PSMM iterative framework can be guaranteed by [26] [32], so that we can get the following results:

**Definition 1.** For given parameters  $C \in \mathbb{R}$  and  $\tau \in \mathbb{R}_{++}$ , we say  $((\mathbf{W}^*, b^*), \mathbf{S}^*)$  is the proximal stationary (P-stationary) point of (5) if there exists a Lagrangian multiplier  $\mathbf{\Lambda}^* \in \mathbb{R}^{l \times m}$  such that

$$0 = \mathbf{W}^* - C \sum_{i=1}^n y_i \mathbf{X}_i \left[ 1 - y_i (\langle \mathbf{W}^*, \mathbf{X}_i \rangle + b^*) \right] - \mathbf{\Lambda}^* \tag{15a}$$

$$0 = b^* - C \sum_{i=1}^n y_i \left[ 1 - y_i (\langle \mathbf{W}^*, \mathbf{X}_i \rangle + b^*) \right] \tag{15b}$$

$$0 \in \tau \partial \|\mathbf{S}^*\|_* + \mathbf{\Lambda}^* \tag{15c}$$

$$0 = \mathbf{S}^* - \mathbf{W}^* \tag{15d}$$

**Theorem 2.** Suppose  $((\mathbf{W}^*, b^*), \mathbf{S}^*, \mathbf{\Lambda}^*)$  be the limit point of the sequence  $((\mathbf{W}^k, b^k), \mathbf{S}^k, \mathbf{\Lambda}^k)$  generated by the PSMM iterative framework, then  $((\mathbf{W}^*, b^*), \mathbf{S}^*)$  is a P-stationary point and thus a locally optimal solution to the problem (5).

**Remark.** Since  $(\mathbf{W}^{k+1}, b^{k+1})$  minimizes  $L_\rho((\mathbf{W}, b), \mathbf{S}^k, \mathbf{\Lambda}^k)$ , then we have the following relation:

$$0 = \mathbf{W}^{k+1} - C \sum_{i=1}^n y_i \mathbf{X}_i \left[ 1 - y_i (\langle \mathbf{W}^{k+1}, \mathbf{X}_i \rangle + b^{k+1}) \right] - \mathbf{\Lambda}^k - \rho (\mathbf{S}^k - \mathbf{W}^{k+1}) \tag{16}$$

$$0 = b^{k+1} - \sum_{i=1}^n y_i \left[ 1 - y_i (\langle \mathbf{W}^{k+1}, \mathbf{X}_i \rangle + b^{k+1}) \right] \tag{17}$$

Similarly, since  $S^{k+1}$  minimizes  $L_\rho((W^{k+1}, b^{k+1}), S, \Lambda^k)$ , then we also have the following relation:

$$0 \in \tau \partial \|S^{k+1}\|_* + \Lambda^k + \rho(S^{k+1} - W^{k+1}) \tag{18}$$

1) It is not difficult to find that the iteration formula (9) always produces a gap  $\rho(S^{k+1} - W^{k+1})$  between the state variables  $\Lambda^k$  and  $\Lambda^{k+1}$  at iteration  $k+1$ , so that there always has a residual between  $\Lambda^k$  and the locally optimum  $\Lambda^*$  on each update. Meanwhile, from the perspective of feasibility, the locally optimal point of (5) must meet the feasible condition (15d). Nevertheless,  $W^{k+1}$  and  $S^{k+1}$  are generally unequal on each update. To this end, we denote  $E_p^{k+1} := S^{k+1} - W^{k+1}$  as the primal residual at iteration  $k+1$ . Additionally, combining (9) and (16), we can obtain the following equality:

$$0 = W^{k+1} - C \sum_{i=1}^n y_i X_i [1 - y_i (\langle W^{k+1}, X_i \rangle + b^{k+1})] - \Lambda^{k+1} + \rho(S^{k+1} - S^k)$$

which means that there always exists a residual  $\rho(S^{k+1} - S^k)$  for the condition (15a) on each iteration. We denote  $E_d^{k+1} := \rho(S^{k+1} - S^k)$  as the dual residual at iteration  $k+1$ .

2) Conversely, by (17), it is easy to know that  $b^{k+1}$  always satisfies the condition (15b) at iteration  $k+1$ . Similarly, combining (9) and (18), we can obtain the following equality:

$$0 \in \tau \partial \|S^{k+1}\|_* + \Lambda^{k+1}$$

which means that  $S^{k+1}$  and  $\Lambda^{k+1}$  also always satisfy the condition (15c) at iteration  $k+1$ .

### 4.3. Stopping Criterion

Based on the above analysis, we know that the PSMM algorithm can obtain the locally optimal solution when the following conditions are satisfied:

$$\begin{aligned} E_p^{k+1} &\rightarrow 0 \text{ as } k \rightarrow \infty, \\ E_d^{k+1} &\rightarrow 0 \text{ as } k \rightarrow \infty. \end{aligned}$$

Therefore, inspired by [26], we set a stopping criterion that the decision variables stop updating and output when

$$\|E_p^{k+1}\|_F \leq \varepsilon_p, \|E_d^{k+1}\|_F \leq \varepsilon_d.$$

The tolerances  $\varepsilon_p, \varepsilon_d > 0$  can be predetermined by the following criterion:

$$\begin{aligned} \varepsilon_p &= \sqrt{n} \varepsilon_{abs} + \varepsilon_{rel} \max \left\{ \|W^{k+1}\|_F, \|S^{k+1}\|_F \right\} \\ \varepsilon_d &= \sqrt{n} \varepsilon_{abs} + \varepsilon_{rel} \rho \|\Lambda^{k+1}\|_F \end{aligned}$$

where  $\varepsilon_{abs}$  and  $\varepsilon_{rel}$  are absolute and relative tolerances, respectively, and their settings depend on the application, for details, see [26]. Furthermore, the specific procedure of PSMM algorithm is shown as follows:

---

**PSMM Algorithm for Binary Classification**


---

**Input:** Training data set  $T_m = \{(\mathbf{X}_i, y_i)\}_{i=1}^n$ ,

the row and column size of matrix samples  $l, m$ ,

penalty parameters  $C, \tau, \rho$  and accelerator multiplier  $\alpha$ .

**Initialize:**  $\mathbf{S}^0 = \mathbf{0} \in \mathbb{R}^{l \times m}$ ,  $\mathbf{\Lambda}^0 = \mathbf{0} \in \mathbb{R}^{l \times m}$ .

for  $k = 0 : \text{max\_step}$

Calculate the optimal multiplier  $\lambda^k$  by the linear system (13)

$$(\mathbf{W}^{k+1}, b^{k+1}) = \left( \frac{1}{1+\rho} (\rho \mathbf{S}^k + \mathbf{\Lambda}^k + \sum_{i=1}^n y_i \lambda_i^k \mathbf{X}_i), \sum_{i=1}^n y_i \lambda_i^k \right);$$

Acceleration operation:  $\mathbf{W}^{k+1} = \alpha \mathbf{W}^{k+1} + (1 - \alpha) \mathbf{S}^k$ ;

$$\mathbf{S}^{k+1} = \mathcal{D}_{\frac{\tau}{\rho}}(\mathbf{W}^{k+1} - \frac{1}{\rho} \mathbf{\Lambda}^k);$$

$$\mathbf{\Lambda}^{k+1} = \mathbf{\Lambda}^k + \rho(\mathbf{S}^{k+1} - \mathbf{W}^{k+1});$$

if  $\|\mathbf{E}_p^{k+1}\|_F \leq \varepsilon_p$ ,  $\|\mathbf{E}_d^{k+1}\|_F \leq \varepsilon_d$

$$\mathbf{W}^* = \mathbf{W}^{k+1}; b^* = b^{k+1};$$

break;

end

$k = k + 1$ ;

end

**Output:**  $\mathbf{W}^*$ ,  $b^*$

---

## 5. Experiments

To verify the binary classification effect of our method, we conduct the experimental analysis of PSMM classifier in this section. There are five selected image data sets, including minst digital database [27], MIT face database, INRIA person database [28], the students face database [29] and Japan female facial expression database (JAFFE) [30]. And, we conduct comparative experiments on these data sets with PSVM [1], TSVM [7], LRSMM [16], LTMRSMM [18]. We summarize the main information of the selected data sets in **Table 1**. Some pictures of these databases are shown in **Figures 1-5**. All experiments are implemented in Matlab R2019a on a workstation with AMD A10-7300 Radeon R6 1.90 Hz, 10 Computer Cores 4C+6G, 4 GB RAM, and 64 bit Windows Server 2009 system.

**Table 1.** Summary of five image data sets.

database	size (pos/neg)	dimension	class number
Minst	5000/5000	28 × 28	10
MIT	2706/4381	20 × 20	2
Students face	200/200	200 × 200	2
INRIA person	2416/12180	64 × 128	2
JAFFE	175/175	256 × 256	7

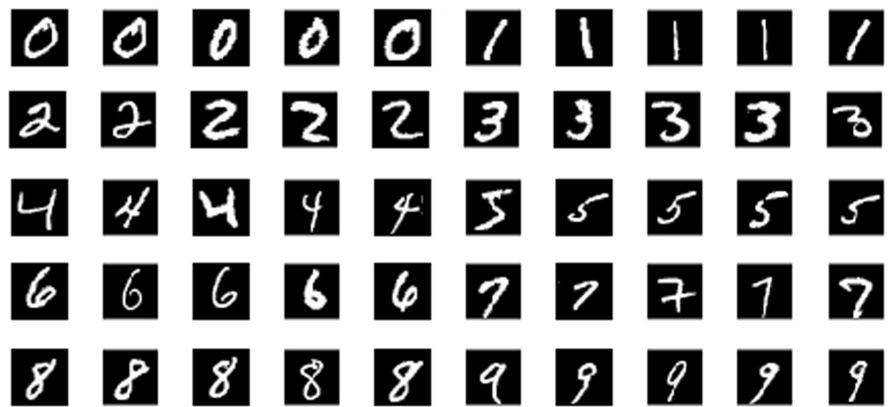


Figure 1. Some pictures from minst digital database.



Figure 2. Some pictures from MIT face database.



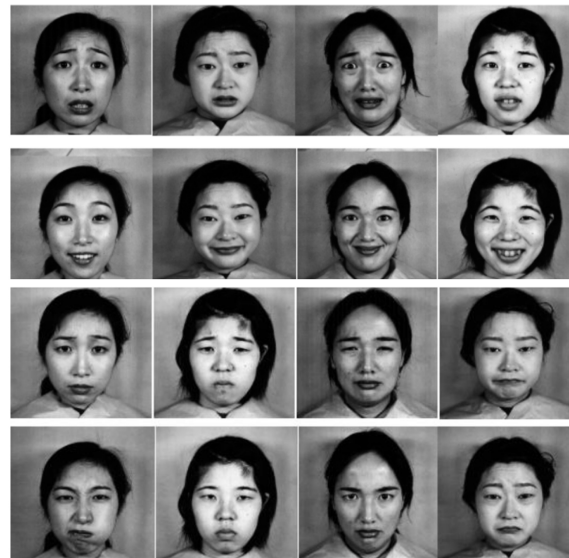
Figure 3. Some pictures from the students face database.

### 5.1. Classification on Minst Digital Database

Minst digital database [27] comes from National Institute of Standards and Technology, which includes handwritten characters from zero to nine. Due to



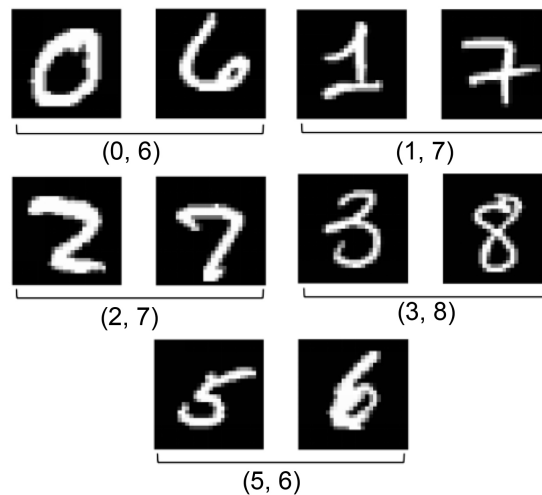
**Figure 4.** Some pictures from the INRIA person database.



**Figure 5.** Some pictures from the JAFFE database.

the differences in writing habits of volunteers, different digits may have similar contours, resulting in wrong judgment. The purpose why we chose this database is to distinguish handwritten numerals with little difference in contour.

Through the observation of a large number of handwritten character samples, we found five digital pairs, and the two numbers of any digital pair are similar in morphology. They are (0, 6), (1, 7), (2, 7), (3, 8) and (5, 6), respectively. And, their comparisons are shown in **Figure 6**. We conducted binary classification experiments on the two numbers in the specific digital pair. The experimental results show that PSMM performs better than PSVM [1], TSVM [7], LRSMM [16] and LTMRSM [18] in distinguishing handwritten digits with similar contours, for details, see **Table 2**.



**Figure 6.** Comparison pictures of handwritten digits with similar contours.

**Table 2.** Experimental results on minst digital database.

Minst	PSVM	TSVM	LRSM	LTMRSM	PSMM
Subject	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)
	Time (s)	Time (s)	Time (s)	Time (s)	Time (s)
(0, 6)	97.23	98.05	97.59	97.40	<b>98.60</b>
	0.09434	0.21151	25.29174	4.79779	0.79307
(1, 7)	98.35	98.87	97.24	98.07	<b>99.07</b>
	0.10088	0.22491	31.38959	5.42440	0.81284
(2, 7)	96.02	97.54	95.80	96.77	<b>97.87</b>
	0.10020	0.20906	23.40696	5.40644	0.69939
(3, 8)	91.71	92.53	91.09	93.12	<b>94.80</b>
	0.09304	0.21207	26.43666	5.07218	0.55883
(5, 6)	94.51	95.79	94.49	95.64	<b>96.66</b>
	0.09083	0.21531	31.84351	5.23974	0.60023

## 5.2. Classification on Portrait Database

Through the results on minst digital database, we found that the novel classifier has a good ability to distinguish two categories with little difference. It prompted us to further apply it to more complex portrait classification tasks, such as face recognition, gender judgment, pedestrian detection and emotion recognition, which require classifiers to have the ability to distinguish details. To test the performance of PSMM classifier in these complex tasks, we selected the following four portrait databases.

1) MIT face database is affiliated to Massachusetts Institute of Technology, with a total of 7087 samples, including 2706 face samples and 4381 no face sam-

ples. All images are gray images stored in BMP format, with width and height of 20. We chose it for face recognition experiments which is to judge whether there is a face in picture.

2) INRIA person database [28] was collected to detect whether there exist people in the image, including 2416 pedestrian pictures and 12,180 landscape pictures. All images are color images stored in JPG or PNG format, and we have normalized the samples into  $61 \times 128$  gray level matrices to unify the specifications of samples.

3) The students face database [29] contains 400 photos of medical students in Stanford University, which consists of 200 males and 200 females. All images are gray images stored in JPG format, with the width and height of 200. In the process of gender judgment, hair style will be a disturbance, because girls may have short hair and boys may have long hair. Therefore, this database we chose will further test the ability of classifiers to distinguish facial details.

4) Japan female facial expression database [30] has 213 facial expression images, which are composed of 7 facial expression images of 10 women. The seven expressions are afraid, surprised, happy, sad, angry, disgusted, neutral, respectively. All images are gray images stored in TIFF format, with the width and height of 256. We conducted binary classification experiments between each emotion and the rest to observe the sensitivity of PSMM classifier to facial expression differences.

Through a large number of preliminary experiments, we have determined the appropriate training set size of each database. And, the stability and progressiveness of the novel algorithm have been verified by properly increasing the number of test samples and massive repeated experiments, for details, see **Tables 3-6**.

**Table 3.** Experimental results on MIT face database.

MIT	PSVM	TSVM	LRSMM	LTMRSM	PSMM
Test set	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)
pos/neg	Time (s)	Time (s)	Time (s)	Time (s)	Time (s)
100/100	87.70	85.95	89.10	87.00	<b>92.60</b>
	0.00678	0.01823	9.61162	1.53774	0.09710
200/200	89.10	86.15	88.80	85.28	<b>92.15</b>
	0.00753	0.01756	10.32216	1.69821	0.09709
300/300	90.10	87.45	90.07	88.38	<b>92.93</b>
	0.00455	0.01756	10.73969	1.70013	0.09879
400/400	88.09	84.64	88.45	85.65	<b>92.72</b>
	0.00712	0.01797	10.17160	1.65579	0.09469
500/500	90.28	88.73	91.37	87.36	<b>93.61</b>
	0.00705	0.01726	10.37729	1.61703	0.10190

**Table 4.** Experimental results on INRIA person database.

INRIA	PSVM	TSVM	LRSMM	LTMRSMM	PSMM
Test set	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)
pos/neg	Time (s)	Time (s)	Time (s)	Time (s)	Time (s)
100/100	79.60	83.00	82.00	72.55	<b>86.85</b>
	0.53785	1.36313	19.32664	4.96127	3.46110
200/200	79.73	82.45	80.53	72.40	<b>87.20</b>
	0.54470	1.36510	20.32056	5.21338	3.82192
300/300	80.02	84.25	82.32	75.28	<b>87.23</b>
	0.55377	1.37249	20.67401	5.15255	3.72906
400/400	79.39	82.76	79.24	73.60	<b>86.57</b>
	0.68570	1.73248	26.24048	6.93018	4.62039
500/500	79.44	81.91	79.18	74.27	<b>85.67</b>
	0.70342	1.75017	31.68549	7.36364	5.26839

**Table 5.** Experimental results on the students face database.

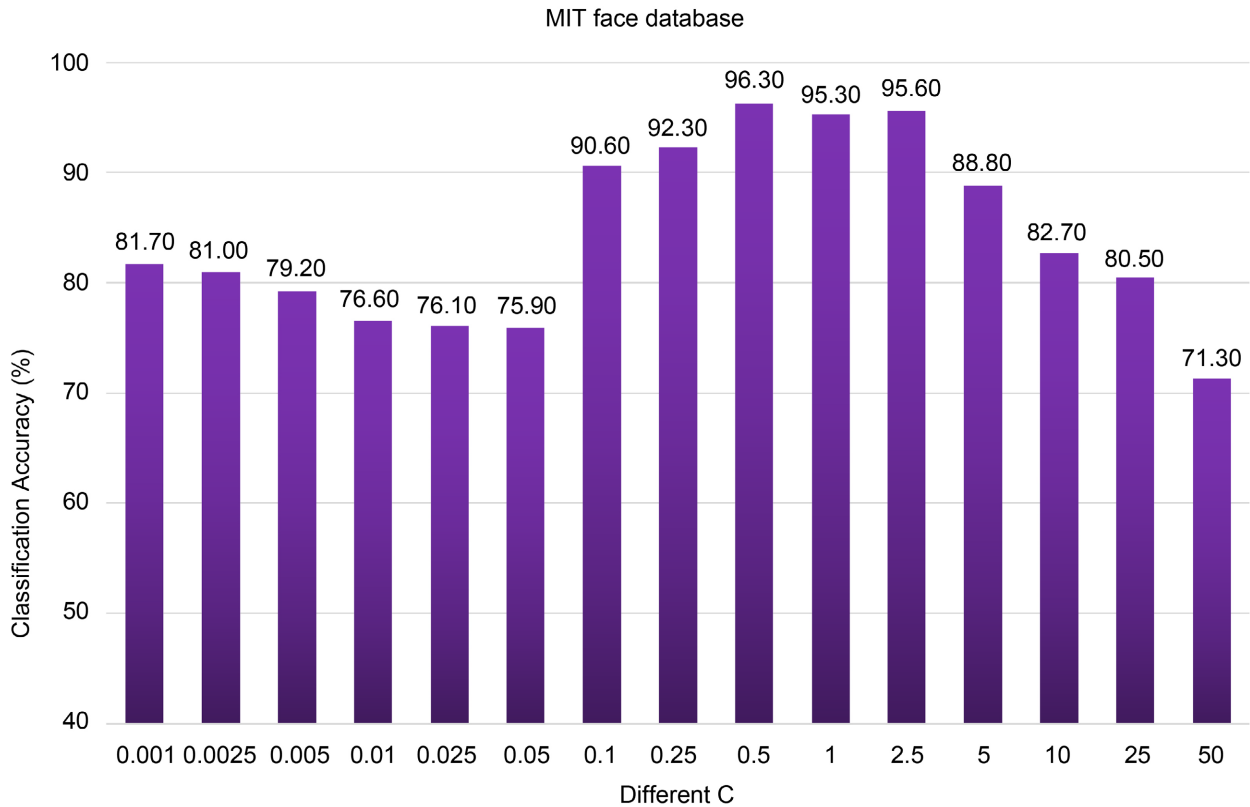
Students	PSVM	TSVM	LRSMM	LTMRSMM	PSMM
Test set	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)
pos/neg	Time (s)	Time (s)	Time (s)	Time (s)	Time (s)
40/40	90.13	90.63	88.63	80.25	<b>94.25</b>
	0.67461	1.69127	5.29848	1.17995	4.77109
50/50	88.20	88.20	88.00	80.00	<b>92.60</b>
	0.69538	1.65039	5.30147	1.16947	5.33867
60/60	88.42	88.83	87.67	79.92	<b>93.00</b>
	0.67033	1.66927	5.50830	1.16938	4.78024
70/70	88.86	88.79	87.86	78.57	<b>93.07</b>
	0.66228	1.61085	5.45025	1.15839	4.82768
80/80	90.69	91.44	90.32	78.00	<b>93.88</b>
	0.65905	1.61492	5.69662	1.10800	5.03019

### 5.3. Parameters Selection

The penalty parameter  $C$  affects the fitting degree of the proximal plane to samples. If the value of  $C$  is too small, the prediction ability of proximal planes will be lost, resulting in under fitting. Conversely,  $C$  with too large value will produce over fitting phenomenon, which will lead to the poor generalization ability of

**Table 6.** Experimental results on the JAFFE database.

JAFFE	PSVM	TSVM	LRSMM	LTMRSMM	PSMM
Subject	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)	Accuracy (%)
	Time (s)	Time (s)	Time (s)	Time (s)	Time (s)
afraid	66.73	82.50	78.44	79.06	<b>86.41</b>
	2.59157	5.67755	4.84327	0.77936	11.52640
surprised	71.51	84.51	83.67	84.17	<b>90.17</b>
	3.31090	7.07017	6.49447	1.37633	18.93282
happy	61.62	86.30	84.20	82.10	<b>90.17</b>
	3.63068	7.86516	6.38803	1.55475	21.77760
sad	62.43	78.88	72.59	74.68	<b>82.10</b>
	3.63619	7.83720	8.26077	1.52470	16.67137
angry	67.51	87.50	86.34	85.33	<b>90.00</b>
	3.62794	7.90733	7.06966	1.38328	22.54246
disgusted	66.22	86.21	82.07	82.07	<b>88.80</b>
	3.62028	7.88570	8.71498	1.59652	19.36869
neutral	58.01	80.01	76.84	77.83	<b>84.17</b>
	3.63031	7.83308	8.32018	1.51735	17.49704

**Figure 7.** Effects of different C on the performance of PSMM classifier in the task of judging the existence of human face.

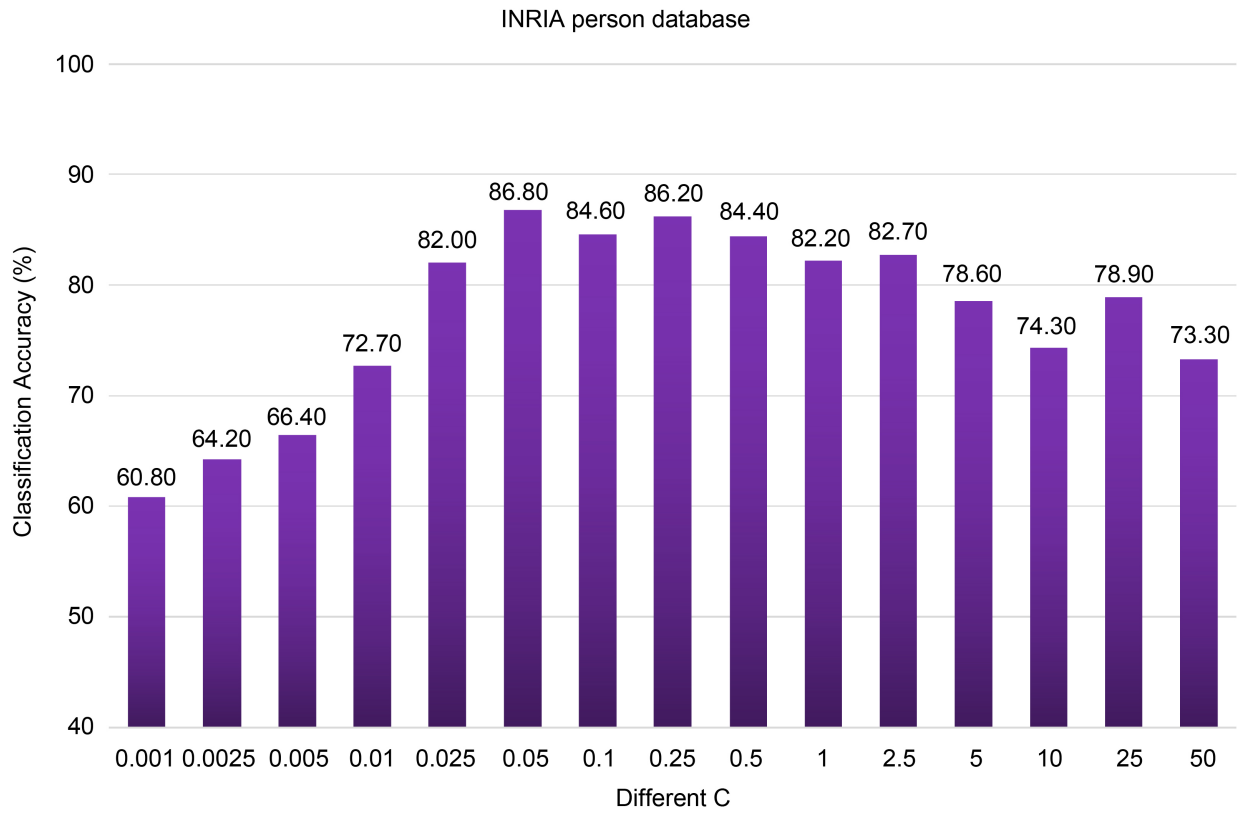


Figure 8. Effects of different C on the performance of PSMM classifier in the task of judging the existence of pedestrian.

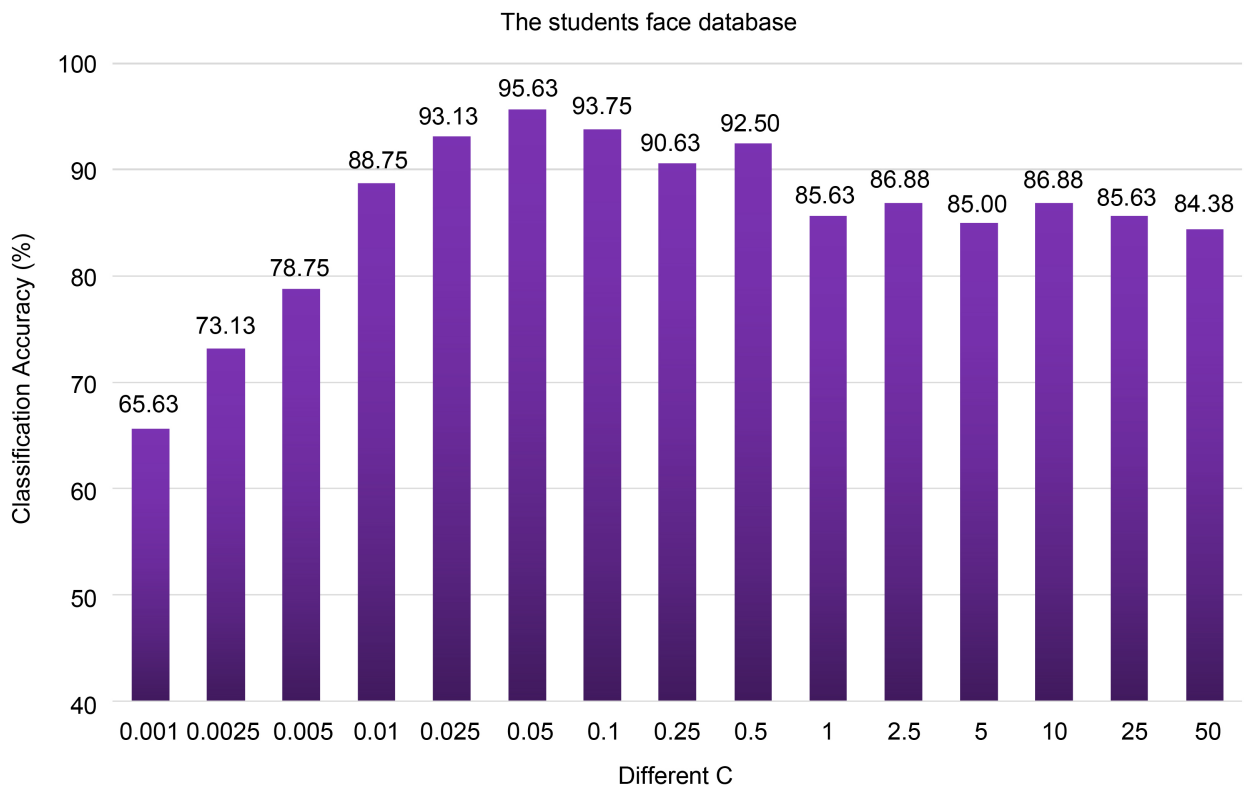
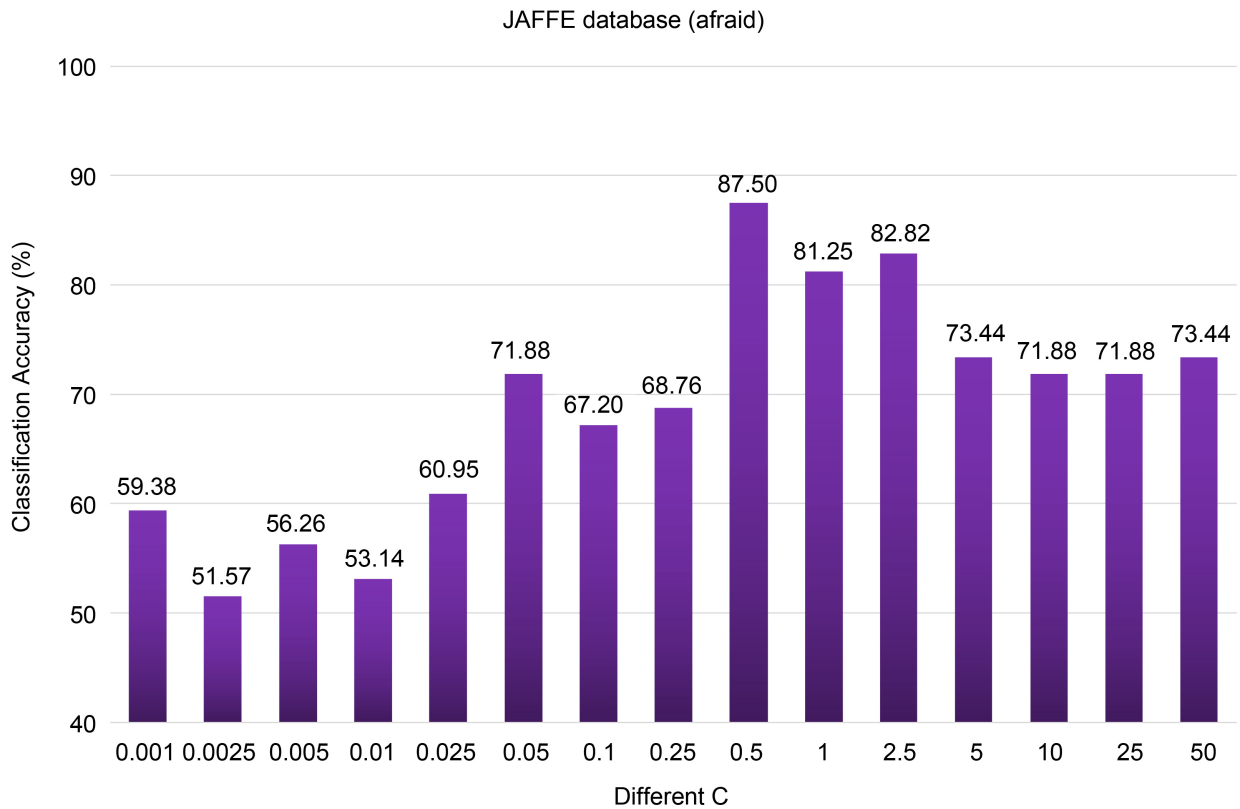
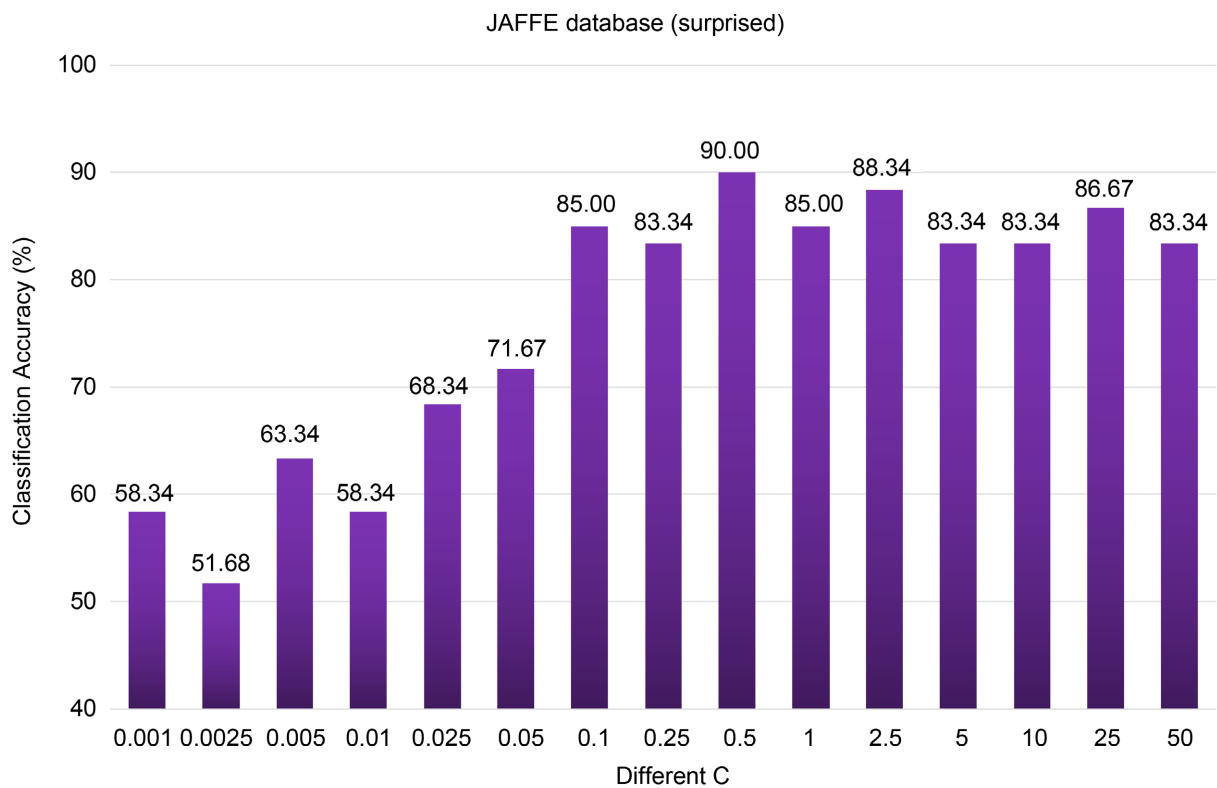


Figure 9. Effects of different C on the performance of PSMM classifier in the task of judging the gender of students.



**Figure 10.** Effects of different C on the performance of PSMM classifier in the task of distinguishing afraid expression.



**Figure 11.** Effects of different C on the performance of PSMM classifier in the task of distinguishing surprised expression.

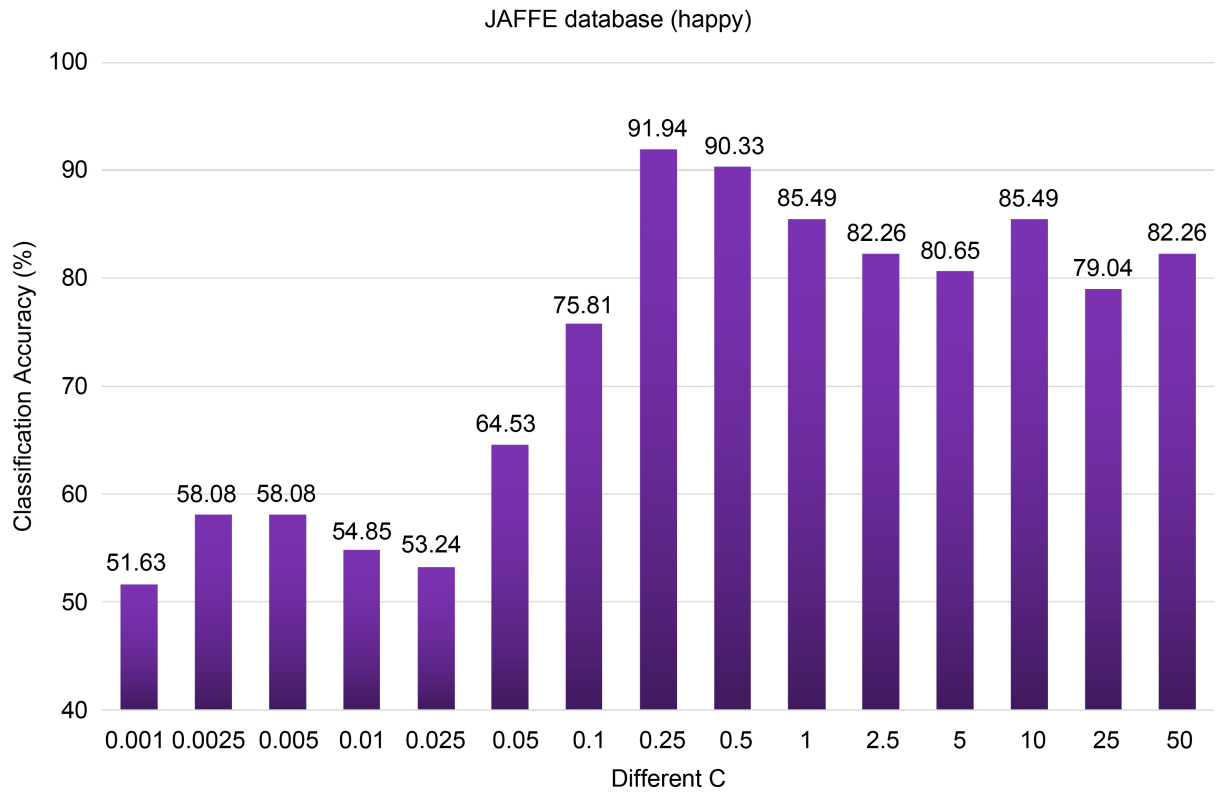


Figure 12. Effects of different C on the performance of PSMM classifier in the task of distinguishing happy expression.

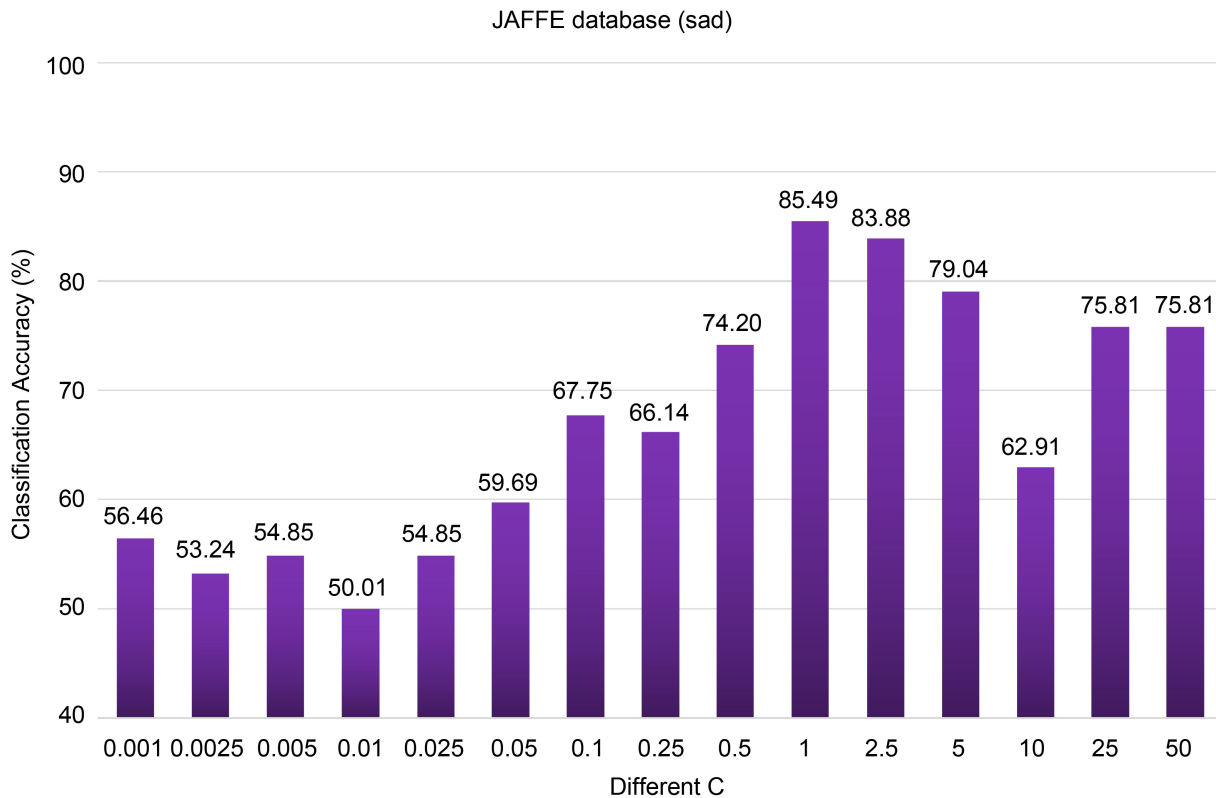
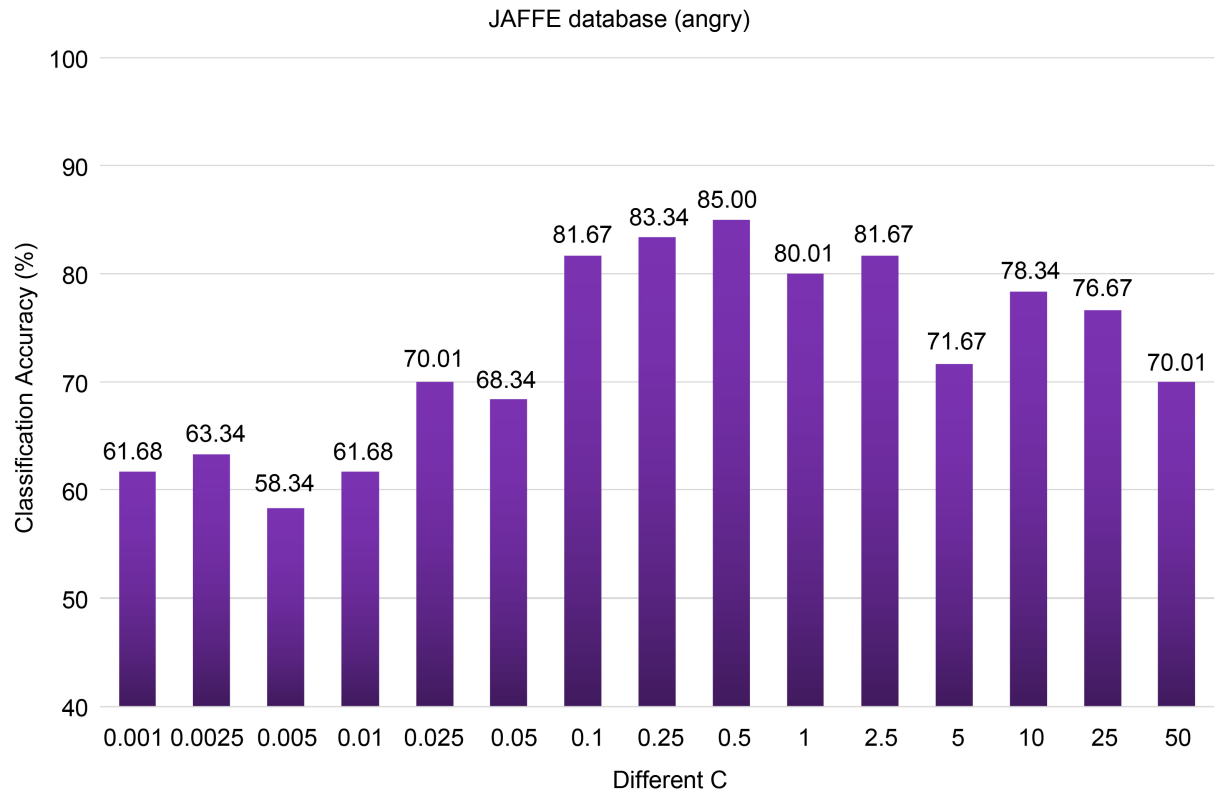
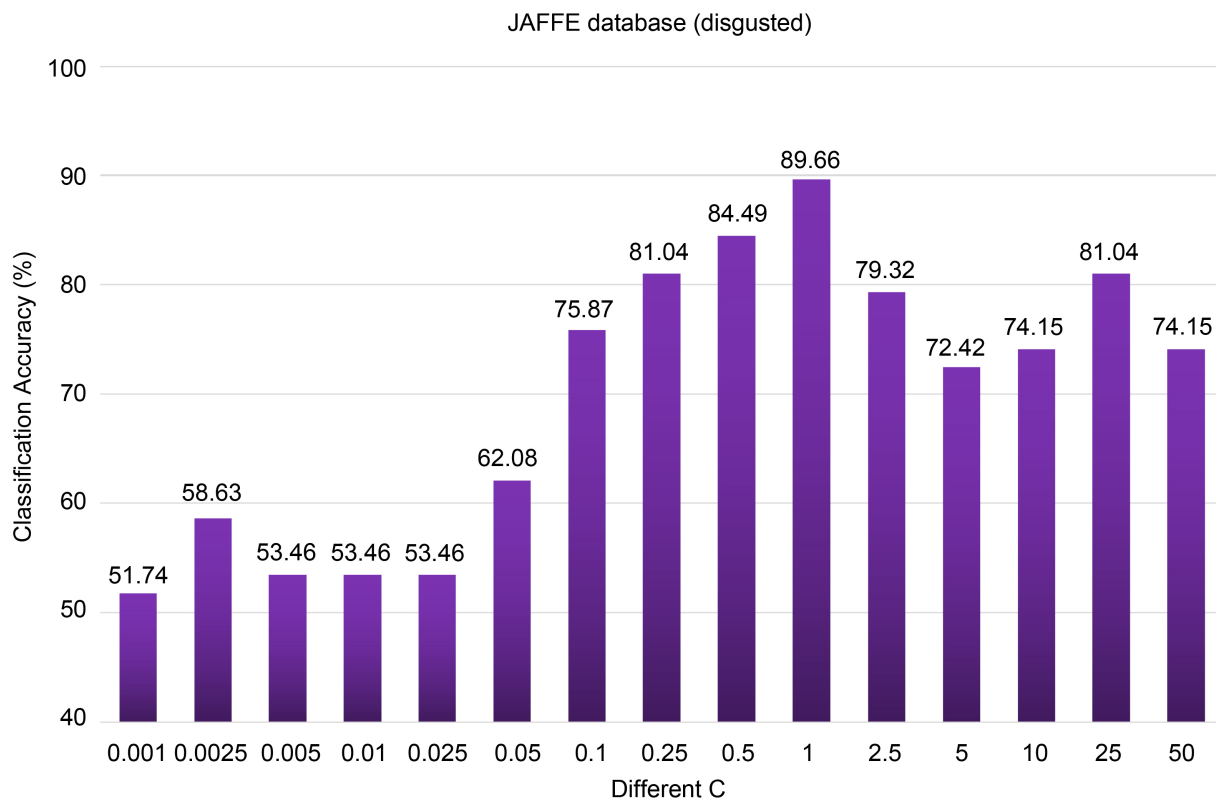


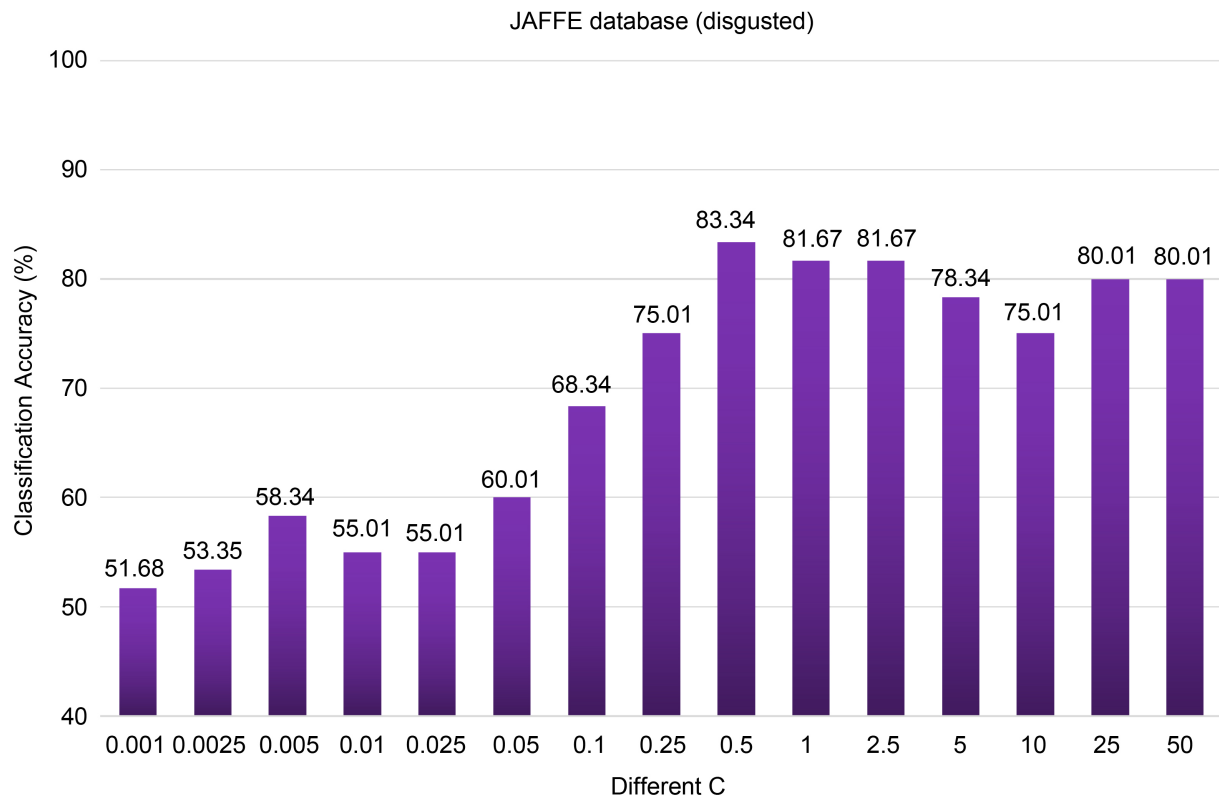
Figure 13. Effects of different C on the performance of PSMM classifier in the task of distinguishing sad expression.



**Figure 14.** Effects of different C on the performance of PSMM classifier in the task of distinguishing angry expression.



**Figure 15.** Effects of different C on the performance of PSMM classifier in the task of distinguishing disgusted expression.



**Figure 16.** Effects of different  $C$  on the performance of PSMM classifier in the task of distinguishing neutral expression.

our model. Thus, predetermining the value of  $C$  is of vital importance for PSMM classifier. To observe the influence of different  $C$  on the classification accuracy of the novel model, we select  $C$  from  $\{1 \times 10^{-3}, 2.5 \times 10^{-3}, 5 \times 10^{-3}, 1 \times 10^{-2}, 2.5 \times 10^{-2}, 5 \times 10^{-2}, \dots, 1 \times 10^1, 2.5 \times 10^1, 5 \times 10^1\}$  and fix the values of other parameters, *i.e.*,  $\tau = 2, \rho = 1$ . The results of different data sets are shown in **Figures 7-16**.

## 6. Conclusion

In this paper, a novel SMM-type method called PSMM is proposed for the matrix classification problems, which has absorbed the advantages of PSVM and LRSMM. In design, the novel method has considered both the relationship between samples within a class and the structure of rows or columns of matrix data, which makes PSMM have good properties to meet the challenges of complex image classification problems. Finally, to verify the performance of our design, we conduct a large number of comparative experiments. It can be seen from the experimental results that PSMM performs better than PSVM, TSVM, LRSMM, LTMRSSMM in the demanding image classification tasks. Moreover, since we only considered the linearly binary classification situation and the selection of penalty parameter  $C$  in this paper, there still have some relevant topics worthy of in-depth study in the future. For example, how to extend the PSMM algorithm to multiple classification situation, how to introduce appropriate kernel function to create

the nonlinear version of PSMM and how to select a stable parameter combination. We will take these topics as our future directions.

## Acknowledgements

Sincerely thank Professor Yulan Liu for the support and guidance to this research, and gratefully acknowledge Guangdong University of Technology for providing a good learning platform.

## Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

## References

- [1] Fung, G. and Mangasarian, O.L. (2001) Proximal Support Vector Machine Classifiers. *Proceedings of Seventh International Conference on Knowledge and Data Discovery*, San Francisco, 26-29 August 2001, 77-86. <https://doi.org/10.1145/502512.502527>
- [2] Mangasarian, O.L. and Wild, E.W. (2006) Multisurface Proximal Support Vector Machine Classification via Generalized Eigenvalues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **28**, 69-74. <https://doi.org/10.1109/TPAMI.2006.17>
- [3] Chen, W.J. and Tian, Y.J. (2010) Lp-Norm Proximal Support Vector Machine and Its Applications. *Procedia Computer Science*, **1**, 2417-2423. <https://doi.org/10.1016/j.procs.2010.04.272>
- [4] Zhu, Z.F., Zhu, X.Q., Guo, Y.F., Ye, Y.D. and Xue, X.Y. (2012) Inverse Matrix-free Incremental Proximal Support Vector Machine. *Decision Support Systems*, **53**, 395-405. <https://doi.org/10.1016/j.dss.2012.02.007>
- [5] Shao, Y.H., Deng, N.Y., Chen, W.J. and Wang, Z. (2013) Improved Generalized Eigenvalue Proximal Support Vector Machine. *IEEE Signal Processing Letters*, **20**, 213-216. <https://doi.org/10.1109/LSP.2012.2216874>
- [6] Li, G.Q., Yang, L.X., Wu, Z.Y. and Wu, C.Z. (2021) D.C. Programming for Sparse Proximal Support Vector Machines. *Information Sciences*, **547**, 187-201. <https://doi.org/10.1016/j.ins.2020.08.038>
- [7] Jayadeva, Khemchandani, R. and Chandra, S. (2007) Twin Support Vector Machines for Pattern Classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **29**, 905-910. <https://doi.org/10.1109/TPAMI.2007.1068>
- [8] Qi, Z.Q., Tiana, Y.J. and Shi, Y. (2012) Laplacian Twin Support Vector Machine for Semi-Supervised Classification. *Neural Networks*, **35**, 46-53. <https://doi.org/10.1016/j.neunet.2012.07.011>
- [9] Guo, J.H., Yi, P., Wang, R.L., Ye, Q.L. and Zhao, C.X. (2014) Feature Selection for Least Squares Projection Twin Support Vector Machine. *Neurocomputing*, **144**, 174-183. <https://doi.org/10.1016/j.neucom.2014.05.040>
- [10] Chen, S.G., Wu, X.J. and Yin, H.F. (2019) A Novel Projection Twin Support Vector Machine for Binary Classification. *Soft Computing*, **23**, 655-668. <https://doi.org/10.1007/s00500-017-2974-z>
- [11] An, Y.X. and Xue, H. (2022) Indefinite Twin Support Vector Machine with DC Functions Programming. *Pattern Recognition*, **121**, Article ID: 108195.

- <https://doi.org/10.1016/j.patcog.2021.108195>
- [12] Mehrkanoon, S., Huang, X.L. and Suykens, J.A.K. (2014) Non-Parallel Support Vector Classifiers with Different Loss Functions. *Neurocomputing*, **143**, 294-301. <https://doi.org/10.1016/j.neucom.2014.05.063>
- [13] Huang, X.L., Shi, L. and Suykens, J.A.K. (2014) Asymmetric Least Squares Support Vector Machine Classifiers. *Computational Statistics and Data Analysis*, **70**, 395-405. <https://doi.org/10.1016/j.csda.2013.09.015>
- [14] Sun, J., Fujita, H., Chen, P. and Li, H. (2016) Dynamic Financial Distress Prediction with Concept Drift Based on Time Weighting Combined with Adaboost Support Vector Machine Ensemble. *Knowledge-Based Systems*, **120**, 4-14. <https://doi.org/10.1016/j.knosys.2016.12.019>
- [15] Wang, H.J., Shao, Y.H., Zhou, S.L., Zhang, C. and Xiu, N.H. (2021) Support Vector Machine Classifier via  $L_{0/1}$  Soft-Margin Loss. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (Early Access). <https://doi.org/10.1109/TPAMI.2021.3092177>
- [16] Luo, L., Xie, Y.B., Zhang, Z.H. and Li, W.J. (2015) Support Matrix Machines. *Proceedings of the 32nd International Conference on Machine Learning*, Vol. 37, Lille, 6-11 July 2015, 938-947.
- [17] Cortes, C. and Vapnik, V. (1995) Support-Vector Networks. *Machine Learning*, **20**, 273-297. <https://doi.org/10.1007/BF00994018>
- [18] Gao, X.Z., Fan, L.Y. and Xu, H.T. (2016) A Novel Method for Classification of Matrix Data Using Twin Multiple Rank SMMs. *Applied Soft Computing*, **48**, 546-562. <https://doi.org/10.1016/j.asoc.2016.07.003>
- [19] Zheng, Q.Q., Zhu, F.Y. and Heng, P.A. (2018) Robust Support Matrix Machine for Single Trial EEG Classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, **26**, 551-562. <https://doi.org/10.1109/TNSRE.2018.2794534>
- [20] Kobayashi, T. and Otsu, N. (2012) Efficient Optimization for Low-Rank Integrated Bilinear Classifiers. *Proceedings of the 12th European Conference on Computer Vision*, Vol. 7573, Florence, 7-13 October 2012, 474-487. [https://doi.org/10.1007/978-3-642-33709-3\\_34](https://doi.org/10.1007/978-3-642-33709-3_34)
- [21] Xu, H.T., Fan, L.Y. and Gao, X.Z. (2015) Projection Twin SMMs for 2D Image Data Classification. *Neural Computing and Applications*, **26**, 91-100. <https://doi.org/10.1007/s00521-014-1700-3>
- [22] Zheng, Q.Q., Zhu, F.Y., Qin, J., Chen, B.D. and Heng, P.A. (2017) Sparse Support Matrix Machine. *Pattern Recognition*, 1-12.
- [23] Jiang, R. and Yang, Z.X. (2018) Multiple Rank Multi-Linear Twin Support Matrix Classification Machine. *Journal of Intelligent and Fuzzy Systems*, **35**, 5741-5754. <https://doi.org/10.3233/JIFS-17414>
- [24] Pan, H.Y., Yang, Y., Zheng, J.D., Li, X. and Cheng, J.S. (2020) Symplectic Interactive Support Matrix Machine and Its Application in Roller Bearing Condition Monitoring. *Neurocomputing*, **398**, 1-10. <https://doi.org/10.1016/j.neucom.2020.01.074>
- [25] Li, X., Yang, Y., Pan, H.Y., Cheng J. and Cheng, J.S. (2020) Non-Parallel Least Squares Support Matrix Machine for Rolling Bearing Fault Diagnosis. *Mechanism and Machine Theory*, **145**, Article ID: 103676. <https://doi.org/10.1016/j.mechmachtheory.2019.103676>
- [26] Boyd, S., Parikh, N., Chu, E., Peleato, B. and Eckstein, J. (2011) Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, **3**, 1-122. <https://doi.org/10.1561/22000000016>

- 
- [27] Lecun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998) Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, **86**, 2278-2324. <https://doi.org/10.1109/5.726791>
- [28] Dalal, N. and Triggs, B. (2005) Histograms of Oriented Gradients for Human Detection. *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, San Diego, 20-25 June 2005, 886-893. <https://doi.org/10.1109/CVPR.2005.177>
- [29] Nazir, M., Ishtiaq, M., Batool, A., Jaffar, M.A. and Mirza, A.M. (2010) Feature Selection for Efficient Gender Classification. *Proceedings of the 11th WSEAS International Conference on Neural Networks, Evolutionary Computing and Fuzzy Systems*, Iasi, 13-15 June 2010, 70-75.
- [30] Lyonsa, M., Akamatsu, S., Kamachia, M. and Gyoba, J. (1998) Coding Facial Expressions with Gabor Wavelets. *3rd IEEE International Conference on Automatic Face and Gesture Recognition*, Nara, 14-16 April 1998, 200-205. <https://doi.org/10.1109/AFGR.1998.670949>
- [31] Cai, J.F., Candes, E.J. and Shen, Z.W. (2010) A Singular Value Thresholding Algorithm for Matrix Completion. *SIAM Journal on Optimization*, **20**, 1956-1982. <https://doi.org/10.1137/080738970>
- [32] Chen, L., Sun, D.F. and Toh, K.C. (2015) A Note on the Convergence of ADMM for Linearly Constrained Convex Optimization Problems. *Computational Optimization and Applications*, **66**, 327-343. <https://doi.org/10.1007/s10589-016-9864-7>