

Article

Improved Nonparallel Support Vector Machine for Pattern Classification

Shujun Lian * and Jingjing Yang

College of Management, Qufu Normal University, Rizhao 276826, China; 18963348298@163.com

* Correspondence: lsjsd2003@qfnu.edu.cn

Abstract

In this paper, we propose a new nonparallel support vector machine for binary classification problems and name it the improved nonparallel support vector machine (IMNSVM). The IMNSVM uses a one-sided ε -band and minimizes ε to achieve a better fitting effect for the same class of training points. By introducing a new variable, ρ , the IMNSVM keeps one class of training points at a certain distance from the hyperplane corresponding to another class of training points, keeping them as far away as possible so as to better adapt to the training points and better describe the difference in data distribution between different categories. The IMNSVM can degenerate into the standard support vector machine (SVM) under certain conditions and is applicable to a wider range of data types. Finally, numerical experiments also explain the effectiveness of the method.

Keywords: machine learning; nonparallel support vector machines; quadratic programming problem; KKT conditions

1. Introduction

Support vector machines (SVMs) [1–3] are some of the most important tools in machine learning. They use statistical and optimization methods to solve machine learning problems and offer advantages for dealing with small-sample machine learning problems. Support vector machines have achieved great success in many fields, such as image segmentation, handwriting recognition, face recognition, text classification, and strip-steel surface defect classification [4–12]. However, in the era of big data, compared to the flourishing development of deep learning and reinforcement learning, research on SVMs has fallen into a slump. Due to the advantages of SVMs, such as simple and clear principles, a solid statistical learning theory foundation, and outstanding practical computational effects, research on theories, models, and algorithms pertaining to SVMs has been ongoing.

Recently, nonparallel support vector machines have attracted a great deal of research attention. Mangasarian [13] proposed the earliest nonparallel classifier and named it the generalized eigenvalue proximal support vector machine (GEPSVM). By solving two related generalized eigenvalue problems, the GEPSVM determines two nonparallel hyperplanes. Jayadeva [14] proposed twin support vector machine (TWSVM). The TWSVM is designed to generate a pair of nonparallel hyperplanes so that each class of points is as close to its corresponding hyperplane as possible and far away from the hyperplane corresponding to another class of points. TWSVMs [14–16] solve two small quadratic programming problems (QPPs) instead of the large quadratic programming problem (QPP) addressed by a traditional support vector machine. Their calculation efficiency is faster, and the effect is more significant. Shao [17] proposed TBSVM, which introduced regularization terms on the



Academic Editor: Ioannis Tsoulos

Received: 26 December 2025

Revised: 25 January 2026

Accepted: 27 January 2026

Published: 3 February 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons](#)[Attribution \(CC BY\) license](#).

basis of TWSVM to reduce structural risk. TBSVM can classify multidimensional complex data better than TWSVM. Tian [18] proposed the nonparallel support vector machine (NPSVM). By adding an ε -band, the NPSVM posed two new small quadratic programming problems, and, like SVMs, it has various advantages, such as sparsity, direct application of the kernel trick in nonlinear cases, and a lack of a requirement for matrix inversion in the calculation process. The study of nonparallel support vector machines has become an important direction of classification research [19–26].

Let us compare support vector machines with nonparallel support vector machines. SVMs are more theoretically mature, but their decision boundaries are relatively rigid, and they demand higher computational complexity for large-scale data. Nonparallel support vector machines can be decomposed into sub-problems for computation, making them more suitable for large-scale data and more robust to noise. They are also suitable for nonparallel data and can be extended to multi-classification problems, making them more suitable for complex distributions and multi-modal problems. In practice, it may be necessary to choose a method through cross-validation verification based on the complexity of the problem, data size, and interpretability requirements. The NSVM proposed by Shao [27] introduced a new soft-interval ε -insensitive loss function for classification problems and fitted the dense regions of each class of training points by minimizing the one-sided ε -band. The NSVM is a nonparallel support vector machine model that can degenerate into the standard SVM under certain conditions. It can achieve conversion between parallel and nonparallel algorithms and be used for a wider range of data classification.

In this paper, we further investigate nonparallel support vector machines, which can degenerate into SVMs, and propose the IMNSVM. We introduce a new variable that is not used in the NSVM: ρ . By adjusting this variable, we can ensure each class of training points is as far as possible from the hyperplane corresponding to another class of training points to improve the separation of training points so that the model has a better classification effect.

This paper is organized as follows. The C-SVM, TBSVM, NPSVM, and NSVM are briefly reviewed in Section 2. We introduce the improved nonparallel support vector machine (IMNSVM) in Section 3. The experimental results are presented in Section 4. In Section 5, the conclusions are given.

2. Background

In this section, we briefly introduce some support vector classifiers. Consider the binary classification problem with the following training set:

$$T = \{(x_1, +1), (x_2, +1), \dots, (x_p, +1), (x_{p+1}, -1), (x_{p+2}, -1), \dots, (x_{p+q}, -1)\}, \quad (1)$$

where $x_i \in R^n, i = 1, 2, \dots, p + q$ are inputs, and $+1, -1$ are outputs.

2.1. C-SVM

In the linear case, C-SVM seeks the separation hyperplane

$$(\omega \cdot x) + b = 0$$

via a QPP as follows:

$$\begin{aligned} \min_{\omega, b, \xi} \quad & \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^{p+q} \xi_i \\ \text{s.t.} \quad & y_i((\omega \cdot x_i) + b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, p + q, \\ & \xi_i \geq 0, \quad i = 1, 2, \dots, p + q, \end{aligned} \quad (2)$$

where $\omega \in R^n$, $y_i \in \{+1, -1\}$, $i = 1, 2, \dots, p + q$, $b \in R$, $C > 0$ is a parameter, and ξ_i , $i = 1, 2, \dots, p + q$ is the slack variable. C-SVM has sparsity and solves its dual problem to implement classification. The kernel trick can be directly applied in the nonlinear case because only the inner products exist in the dual problem.

2.2. TBSVM

Linear TBSVM seeks a pair of nonparallel hyperplanes

$$(\omega_1 \cdot x) + b_1 = 0 \quad \text{and} \quad (\omega_2 \cdot x) + b_2 = 0 \quad (3)$$

by solving two QPPs

$$\begin{aligned} \min_{\omega_1, b_1, \xi_2} \quad & \frac{1}{2} c_1 (\|\omega_1\|^2 + b_1^2) + \frac{1}{2} (A\omega_1 + e_1 b_1)^T (A\omega_1 + e_1 b_1) + c_2 e_2^T \xi_2 \\ \text{s.t.} \quad & (B\omega_1 + e_2 b_1) + \xi_2 \geq e_2, \quad \xi_2 \geq 0, \end{aligned} \quad (4)$$

and

$$\begin{aligned} \min_{\omega_2, b_2, \xi_1} \quad & \frac{1}{2} c_3 (\|\omega_2\|^2 + b_2^2) + \frac{1}{2} (B\omega_2 + e_2 b_2)^T (B\omega_2 + e_2 b_2) + c_4 e_1^T \xi_1 \\ \text{s.t.} \quad & (A\omega_2 + e_1 b_2) + \xi_1 \geq e_1, \quad \xi_1 \geq 0, \end{aligned} \quad (5)$$

where $\omega_k \in R^n$, $b_k \in R$, $k = 1, 2$. $c_i > 0$, $i = 1, 2, 3, 4$ are parameters, $e_1 = (1, 1, \dots, 1)^T \in R^p$, $e_2 = (1, 1, \dots, 1)^T \in R^q$, $A = (x_1, x_2, \dots, x_p)^T \in R^{p \times n}$, $B = (x_{p+1}, x_{p+2}, \dots, x_{p+q})^T \in R^{q \times n}$, and $\xi_1 \in R^p$ and $\xi_2 \in R^q$ are slack variables. TBSVM has difficulty solving the problem posed by an inverse matrix when there are many data. In the nonlinear case, two extra kernel-generated surfaces need be given to replace hyperplanes. Even with linear kernels, the nonlinear classifiers cannot degenerate into linear classifiers.

2.3. NPSVM

The linear NPSVM, given the following two QPPs, seeks two nonparallel hyperplanes (3):

$$\begin{aligned} \min_{\omega_1, b_1, \xi_1, \xi_1^*, \eta_2} \quad & \frac{1}{2} \|\omega_1\|^2 + C_1 \sum_{i=1}^p (\xi_{1i} + \xi_{1i}^*) + C_2 \sum_{j=p+1}^{p+q} \eta_{2j} \\ \text{s.t.} \quad & (\omega_1 \cdot x_i) + b_1 \leq \varepsilon + \xi_{1i}, \quad i = 1, 2, \dots, p, \\ & -(\omega_1 \cdot x_i) - b_1 \leq \varepsilon + \xi_{1i}^*, \quad i = 1, 2, \dots, p, \\ & (\omega_1 \cdot x_j) + b_1 \leq -1 + \eta_{2j}, \quad j = p+1, p+2, \dots, p+q, \\ & \xi_{1i}, \xi_{1i}^* \geq 0, \quad i = 1, 2, \dots, p, \\ & \eta_{2j} \geq 0, \quad j = p+1, p+2, \dots, p+q, \end{aligned} \quad (6)$$

and

$$\begin{aligned} \min_{\omega_2, b_2, \xi_2, \xi_2^*, \eta_1} \quad & \frac{1}{2} \|\omega_2\|^2 + C_3 \sum_{i=p+1}^{p+q} (\xi_{2i} + \xi_{2i}^*) + C_4 \sum_{j=1}^p \eta_{1j} \\ \text{s.t.} \quad & (\omega_2 \cdot x_i) + b_2 \leq \varepsilon + \xi_{2i}, \quad i = p+1, p+2, \dots, p+q, \\ & -(\omega_2 \cdot x_i) - b_2 \leq \varepsilon + \xi_{2i}^*, \quad i = p+1, p+2, \dots, p+q, \\ & (\omega_2 \cdot x_j) + b_2 \geq 1 - \eta_{1j}, \quad j = 1, 2, \dots, p, \\ & \xi_{2i}, \xi_{2i}^* \geq 0, \quad i = p+1, p+2, \dots, p+q, \\ & \eta_{1j} \geq 0, \quad j = 1, 2, \dots, p, \end{aligned} \quad (7)$$

where x_i , $i = 1, 2, \dots, p$ are positive points; x_j , $j = p+1, p+2, \dots, p+q$ are negative points; $C_i > 0$, $i = 1, 2, 3, 4$ are parameters; and $\xi_1 = (\xi_{11}, \xi_{12}, \dots, \xi_{1p})^T$, $\xi_1^* = (\xi_{11}^*, \xi_{12}^*, \dots, \xi_{1p}^*)^T$, $\xi_2 = (\xi_{2,p+1}, \xi_{2,p+2}, \dots, \xi_{2,p+q})^T$, $\xi_2^* = (\xi_{2,p+1}^*, \xi_{2,p+2}^*, \dots, \xi_{2,p+q}^*)^T$, $\eta_1 = (\eta_{11}, \eta_{12}, \dots, \eta_{1p})^T$, and $\eta_2 = (\eta_{2,p+1}, \eta_{2,p+2}, \dots, \eta_{2,p+q})^T$ are slack variables. Taking the positive points as an

example, this model requires as many positive points as possible between the hyperplanes $(\omega_1 \cdot x) + b_1 = \varepsilon$ and $(\omega_1 \cdot x) + b_1 = -\varepsilon$.

2.4. NSVM

The NSVM introduces a new soft-interval ε -insensitive loss function for classification problems:

$$l(x, y, g_k) = \begin{cases} 1 - y(w_k^T x + b_k), & y(w_k^T x + b_k) \leq 1, \\ 0, & 1 \leq y(w_k^T x + b_k) \leq 1 + \varepsilon, \\ y(w_k^T x + b_k) - (1 + \varepsilon), & \text{else.} \end{cases}$$

where $k = 1, 2$, $x \in R^n$ are inputs, and $y \in \{+1, -1\}$ are outputs.

In the linear case, the NSVM solves two primal problems:

$$\begin{aligned} \min_{\omega_1, b_1, \varepsilon_1, \xi_1^{(*)}, \eta_2} \quad & \frac{1}{2} \|\omega_1\|^2 + \gamma \varepsilon_1 + C_1 \sum_{i=1}^p (\xi_{1i} + \xi_{1i}^*) + C_2 \sum_{j=p+1}^{p+q} \eta_{2j} \\ \text{s.t.} \quad & 1 - \xi_{1i}^* \leq (\omega_1 \cdot x_i) + b_1 \leq 1 + \varepsilon_1 + \xi_{1i}, \quad i = 1, 2, \dots, p, \\ & (\omega_1 \cdot x_j) + b_1 \leq -1 + \eta_{2j}, \quad j = p+1, p+2, \dots, p+q, \\ & \xi_{1i}, \xi_{1i}^* \geq 0, \quad i = 1, 2, \dots, p, \\ & \eta_{2j} \geq 0, \quad j = p+1, p+2, \dots, p+q, \\ & \varepsilon_1 \geq 0, \end{aligned}$$

and

$$\begin{aligned} \min_{\omega_2, b_2, \varepsilon_2, \xi_2^{(*)}, \eta_1} \quad & \frac{1}{2} \|\omega_2\|^2 + \gamma \varepsilon_2 + C_1 \sum_{i=1}^p \eta_{1i} + C_2 \sum_{j=p+1}^{p+q} (\xi_{2j} + \xi_{2j}^*) \\ \text{s.t.} \quad & (\omega_2 \cdot x_i) + b_2 \geq 1 - \eta_{1i}, \quad i = 1, 2, \dots, p, \\ & -1 - \varepsilon_2 - \xi_{2j}^* \leq (\omega_2 \cdot x_j) + b_2 \leq -1 + \xi_{2j}, \quad j = p+1, p+2, \dots, p+q, \\ & \eta_{1i} \geq 0, \quad i = 1, 2, \dots, p, \\ & \xi_{2j}, \xi_{2j}^* \geq 0, \quad j = p+1, p+2, \dots, p+q, \\ & \varepsilon_2 \geq 0, \end{aligned}$$

where $C_i > 0, i = 1, 2$ are the parameters. $\xi_1^{(*)} = (\xi_1^T, \xi_1^{*T})^T = (\xi_{11}, \dots, \xi_{1p}, \xi_{11}^*, \dots, \xi_{1p}^*)^T$, $\xi_2^{(*)} = (\xi_2^T, \xi_2^{*T})^T = (\xi_{2,p+1}, \dots, \xi_{2,p+q}, \xi_{2,p+1}^*, \dots, \xi_{2,p+q}^*)^T$, $\eta_1 = (\eta_{11}, \eta_{12}, \dots, \eta_{1p})^T$ and $\eta_2 = (\eta_{2,p+1}, \eta_{2,p+2}, \dots, \eta_{2,p+q})^T$ are slack variables.

For the positive points, the margin between the hyperplanes $(\omega_1 \cdot x) + b_1 = 1$ and $(\omega_1 \cdot x) + b_1 = -1$ can be expressed as $\frac{2}{\|\omega_1\|}$. And $\frac{\varepsilon_1}{\|\omega_1\|}$ is defined as the width of the one-sided ε -band between the hyperplanes $(\omega_1 \cdot x) + b_1 = 1$ and $(\omega_1 \cdot x) + b_1 = 1 + \varepsilon_1$. By minimizing $\|\omega_1\|$, the negative points will be as far away as possible from the hyperplane corresponding to the positive points. Moreover, minimizing ε_1 can reveal a one-sided ε -band that is a better fit for the dense regions of the positive points.

3. IMNSVM

In this section, a new improved nonparallel support vector machine (IMNSVM) is proposed for binary classification with the training set (1).

3.1. Linear IMNSVM

3.1.1. The Original Problems

We look for a pair of nonparallel hyperplanes

$$(\omega_1 \cdot x) + b_1 = 1 \quad \text{and} \quad (\omega_2 \cdot x) + b_2 = -1 \quad (8)$$

by solving two QPPs,

$$\begin{aligned}
 \min_{\omega_1, b_1, \varepsilon_1, \xi_1^{(*)}, \eta_2, \rho_2} \quad & \frac{1}{2} (\|\omega_1\|^2 + b_1^2) + r_1 \varepsilon_1 + C_1 \sum_{i=1}^p (\xi_{1i} + \xi_{1i}^*) - r_2 \rho_2 + C_2 \sum_{j=p+1}^{p+q} \eta_{2j} \\
 \text{s.t.} \quad & (\omega_1 \cdot x_i) + b_1 \leq 1 + \varepsilon_1 + \xi_{1i}, \quad i = 1, 2, \dots, p, \\
 & -(\omega_1 \cdot x_i) - b_1 \leq -1 + \xi_{1i}^*, \quad i = 1, 2, \dots, p, \\
 & (\omega_1 \cdot x_j) + b_1 \leq -\rho_2 + \eta_{2j}, \quad j = p+1, p+2, \dots, p+q, \\
 & \xi_{1i} \geq 0, \xi_{1i}^* \geq 0, \quad i = 1, 2, \dots, p, \\
 & \eta_{2j} \geq 0, \quad j = p+1, p+2, \dots, p+q, \\
 & \varepsilon_1 \geq 0, \rho_2 \geq 0,
 \end{aligned} \tag{9}$$

and

$$\begin{aligned}
 \min_{\omega_2, b_2, \varepsilon_2, \xi_2^{(*)}, \eta_1, \rho_1} \quad & \frac{1}{2} (\|\omega_2\|^2 + b_2^2) + r_3 \varepsilon_2 + C_3 \sum_{j=p+1}^{p+q} (\xi_{2j} + \xi_{2j}^*) - r_4 \rho_1 + C_4 \sum_{i=1}^p \eta_{1i} \\
 \text{s.t.} \quad & -(\omega_2 \cdot x_j) - b_2 \leq 1 + \varepsilon_2 + \xi_{2j}, \quad j = p+1, p+2, \dots, p+q, \\
 & (\omega_2 \cdot x_j) + b_2 \leq -1 + \xi_{2j}^*, \quad j = p+1, p+2, \dots, p+q, \\
 & -(\omega_2 \cdot x_i) - b_2 \leq -\rho_1 + \eta_{1i}, \quad i = 1, 2, \dots, p, \\
 & \xi_{2j} \geq 0, \xi_{2j}^* \geq 0, \quad j = p+1, p+2, \dots, p+q, \\
 & \eta_{1i} \geq 0, \quad i = 1, 2, \dots, p, \\
 & \varepsilon_2 \geq 0, \rho_1 \geq 0,
 \end{aligned} \tag{10}$$

where $C_i > 0$ and $r_i > 0, i = 1, 2, 3, 4$ are parameters, and $x_i, i = 1, 2, \dots, p$ are positive points. $x_j, j = p+1, p+2, \dots, p+q$ are negative points, and $\eta_1 = (\eta_{11}, \eta_{12}, \dots, \eta_{1p})^T$, $\eta_2 = (\eta_{2,p+1}, \eta_{2,p+2}, \dots, \eta_{2,p+q})^T$, $\xi_1^{(*)} = (\xi_1^T, \xi_1^{*T})^T = (\xi_{11}, \dots, \xi_{1p}, \xi_{11}^*, \dots, \xi_{1p}^*)^T$ and $\xi_2^{(*)} = (\xi_2^T, \xi_2^{*T})^T = (\xi_{2,p+1}, \dots, \xi_{2,p+q}, \xi_{2,p+1}^*, \dots, \xi_{2,p+q}^*)^T$ are slack variables.

As with TBSVM, by introducing the transformation from R^n to R^{n+1} , $\frac{1+\rho}{\sqrt{\|\omega_1\|^2 + b_1^2}}$ is defined as the margin between the proximal hyperplane $(\omega_1 \cdot x) + b_1 = [\omega_1^T, b_1][x^T, 1]^T = 1$ and the bounding hyperplane $(\omega_1 \cdot x) + b_1 = [\omega_1^T, b_1][x^T, 1]^T = -\rho$.

For problems (9) and (10), there are five parts in the objective function. The regularization term, serving as the first part, can ensure that structural risk is minimized. The second part minimizes ε to achieve a better fitting effect for the same class of training points. The third and fifth parts minimize the loss as much as possible. The first and fourth parts ensure that each class of training points remains a certain distance from the hyperplane corresponding to another class of training points, ensuring this distance is as large as possible. When $r_1 = r_2 = 0, \rho_2 = 1, C_1 = C_2$, problem (9) can degenerate into the standard SVM. When $r_3 = r_4 = 0, \rho_1 = 1, C_3 = C_4$, problem (10) can degenerate into the standard SVM too.

In geometry, we consider problem (9) in terms of R^2 (Figure 1). Firstly, we hope that as many and as dense positive points “+” as possible fall into the one-sided ε -band between the hyperplanes $(\omega_1 \cdot x) + b_1 = 1$ (red thin solid line) and $(\omega_1 \cdot x) + b_1 = 1 + \varepsilon_1$ (red solid line), and ε is made as small as possible to achieve a better fitting effect. Secondly, we hope the negative points “*” are below the hyperplane $(\omega_1 \cdot x) + b_1 = -\rho_2$ (red dotted line). The distance between the two is made as large as possible by maximizing ρ_2 . Likewise, problem (10) can be represented by three blue lines.

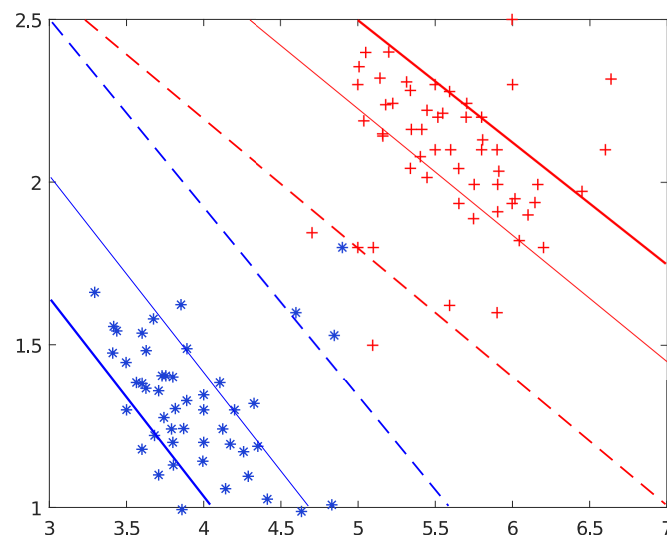


Figure 1. Geometrical illustration of IMNSVM in R^2 .

3.1.2. The Dual Problems

To solve problems (9) and (10), we should derive their dual problems. The Lagrangian function of problem (9) is given as follows:

$$\begin{aligned}
 L(\omega_1, b_1, \varepsilon_1, \xi_1^*, \xi_1, \rho_2, \eta_2, \alpha_1^*, \alpha_1, \beta_2, \lambda_1^*, \lambda_1, \gamma_2, \theta_1, \theta_2) \\
 = \frac{1}{2}(\|\omega_1\|^2 + b_1^2) + r_1\varepsilon_1 + C_1 \sum_{i=1}^p (\xi_{1i}^* + \xi_{1i}) - r_2\rho_2 + C_2 \sum_{j=p+1}^{p+q} \eta_{2j} \\
 + \sum_{i=1}^p \alpha_{1i}^* ((\omega_1 \cdot x_i) + b_1 - 1 - \varepsilon_1 - \xi_{1i}^*) + \sum_{i=1}^p \alpha_{1i} (-(\omega_1 \cdot x_i) - b_1 + 1 - \xi_{1i}) \\
 + \sum_{j=p+1}^{p+q} \beta_{2j} ((\omega_1 \cdot x_j) + b_1 + \rho_2 - \eta_{2j}) - \sum_{i=1}^p \lambda_{1i}^* \xi_{1i}^* - \sum_{i=1}^p \lambda_{1i} \xi_{1i} \\
 - \sum_{j=p+1}^{p+q} \gamma_{2j} \eta_{2j} - \theta_1 \varepsilon_1 - \theta_2 \rho_2,
 \end{aligned} \quad (11)$$

where $\alpha_1^* = (\alpha_{11}^*, \alpha_{12}^*, \dots, \alpha_{1p}^*)^T$, $\alpha_1 = (\alpha_{11}, \alpha_{12}, \dots, \alpha_{1p})^T$, $\beta_2 = (\beta_{2,p+1}, \beta_{2,p+2}, \dots, \beta_{2,p+q})^T$, $\lambda_1^* = (\lambda_{11}^*, \lambda_{12}^*, \dots, \lambda_{1p}^*)^T$, $\lambda_1 = (\lambda_{11}, \lambda_{12}, \dots, \lambda_{1p})^T$, $\gamma_2 = (\gamma_{2,p+1}, \gamma_{2,p+2}, \dots, \gamma_{2,p+q})^T$, $\theta_1 \in R$, and $\theta_2 \in R$ are the Lagrange multipliers. Applying the Karush–Kuhn–Tucker (KKT) conditions [28] leaves us with

$$\nabla_{\omega_1} L = \omega_1 + \sum_{i=1}^p \alpha_{1i}^* x_i - \sum_{i=1}^p \alpha_{1i} x_i + \sum_{j=p+1}^{p+q} \beta_{2j} x_j = 0, \quad (12)$$

$$\nabla_{b_1} L = b_1 + \sum_{i=1}^p \alpha_{1i}^* - \sum_{i=1}^p \alpha_{1i} + \sum_{j=p+1}^{p+q} \beta_{2j} = 0, \quad (13)$$

$$\nabla_{\varepsilon_1} L = r_1 - \sum_{i=1}^p \alpha_{1i}^* - \theta_1 = 0, \quad (14)$$

$$\nabla_{\xi_1^*} L = C_1 e_1 - \alpha_1^* - \lambda_1^* = 0, \quad (15)$$

$$\nabla_{\xi_1} L = C_1 e_1 - \alpha_1 - \lambda_1 = 0, \quad (16)$$

$$\nabla_{\rho_2} L = -r_2 + \sum_{j=p+1}^{p+q} \beta_{2j} - \theta_2 = 0, \quad (17)$$

$$\nabla_{\eta_2} L = C_2 e_2 - \beta_2 - \gamma_2 = 0. \quad (18)$$

$$\alpha_1^*, \alpha_1, \lambda_1^*, \lambda_1 \geq 0, \quad (19)$$

$$\beta_2, \gamma_2 \geq 0, \quad (20)$$

$$\theta_1, \theta_2 \geq 0. \quad (21)$$

From (12) and (13), we find that

$$\omega_1 = \sum_{i=1}^p (\alpha_{1i} - \alpha_{1i}^*) x_i - \sum_{j=p+1}^{p+q} \beta_{2j} x_j, \quad (22)$$

$$b_1 = \sum_{i=1}^p (\alpha_{1i} - \alpha_{1i}^*) - \sum_{j=p+1}^{p+q} \beta_{2j}. \quad (23)$$

From (14)–(21), we find that

$$r_1 - \sum_{i=1}^p \alpha_{1i}^* \geq 0, \quad (24)$$

$$-r_2 + \sum_{j=p+1}^{p+q} \beta_{2j} \geq 0, \quad (25)$$

$$0 \leq \alpha_1^*, \alpha_1 \leq C_1 e_1, \quad (26)$$

$$0 \leq \beta_2 \leq C_2 e_2. \quad (27)$$

Through (11), (14)–(18) and (22)–(27), we arrive at the following dual problem pertaining to problem (9):

$$\begin{aligned} \min_{\alpha_1^*, \alpha_1, \beta_2} & \frac{1}{2} \sum_{i=1}^p \sum_{j=1}^p (\alpha_{1i} - \alpha_{1i}^*) (\alpha_{1j} - \alpha_{1j}^*) ((x_i \cdot x_j) + 1) \\ & - \sum_{i=1}^p \sum_{j=p+1}^{p+q} (\alpha_{1i} - \alpha_{1i}^*) \beta_{2j} ((x_i \cdot x_j) + 1) \\ & + \frac{1}{2} \sum_{i=p+1}^{p+q} \sum_{j=p+1}^{p+q} \beta_{2i} \beta_{2j} ((x_i \cdot x_j) + 1) + \sum_{i=1}^p (\alpha_{1i}^* - \alpha_{1i}) \\ \text{s.t.} & \quad r_1 - \sum_{i=1}^p \alpha_{1i}^* \geq 0, \\ & \quad -r_2 + \sum_{j=p+1}^{p+q} \beta_{2j} \geq 0, \\ & \quad 0 \leq \alpha_{1i}^*, \alpha_{1i} \leq C_1, \quad i = 1, 2, \dots, p, \\ & \quad 0 \leq \beta_{2j} \leq C_2, \quad j = p+1, p+2, \dots, p+q. \end{aligned} \quad (28)$$

(28) can be converted into the following problem:

$$\begin{aligned} \min_{\pi} & \quad \frac{1}{2} \pi^T \Lambda \pi + \kappa^T \pi \\ \text{s.t.} & \quad r_1 - \psi^T \pi \geq 0, \\ & \quad -r_2 + \delta^T \pi \geq 0, \\ & \quad 0 \leq \pi \leq C, \end{aligned} \quad (29)$$

where

$$\begin{aligned} \pi &= (\alpha_1^{*T}, \alpha_1^T, \beta_2^T)^T, \\ \kappa &= (e_1^T, -e_1^T, 0e_2^T)^T, \\ \psi &= (e_1^T, 0e_1^T, 0e_2^T)^T, \\ \delta &= (0e_1^T, 0e_1^T, e_2^T)^T, \\ C &= (C_1 e_1^T, C_1 e_1^T, C_2 e_2^T)^T, \end{aligned}$$

and

$$\begin{aligned}\Lambda &= \begin{pmatrix} \Lambda_1 & \Lambda_2 \\ \Lambda_2^T & \Lambda_3 \end{pmatrix}, \\ \Lambda_1 &= \begin{pmatrix} AA^T + e_1 e_1^T & -AA^T - e_1 e_1^T \\ -AA^T - e_1 e_1^T & AA^T + e_1 e_1^T \end{pmatrix}, \\ \Lambda_2 &= \begin{pmatrix} AB^T + e_1 e_2^T \\ -AB^T - e_1 e_2^T \end{pmatrix}, \\ \Lambda_3 &= BB^T + e_2 e_2^T,\end{aligned}$$

where $e_1 = (1, 1, \dots, 1)^T \in R^p$, $e_2 = (1, 1, \dots, 1)^T \in R^q$, $A = (x_1, x_2, \dots, x_p)^T \in R^{p \times n}$, and $B = (x_{p+1}, x_{p+2}, \dots, x_{p+q})^T \in R^{q \times n}$.

For problem (10), as discussed above, we arrive at

$$\omega_2 = \sum_{j=p+1}^{p+q} (\alpha_{2j}^* - \alpha_{2j})x_j + \sum_{i=1}^p \beta_{1i}x_i, \quad (30)$$

$$b_2 = \sum_{j=p+1}^{p+q} (\alpha_{2j}^* - \alpha_{2j}) + \sum_{i=1}^p \beta_{1i}. \quad (31)$$

The dual problem pertaining to problem (10) is given by

$$\begin{aligned}\min_{\alpha_{2j}^*, \alpha_{2j}, \beta_{1i}} & \frac{1}{2} \sum_{i=p+1}^{p+q} \sum_{j=p+1}^{p+q} (\alpha_{2i} - \alpha_{2i}^*)(\alpha_{2j} - \alpha_{2j}^*)((x_i \cdot x_j) + 1) \\ & - \sum_{i=1}^p \sum_{j=p+1}^{p+q} (\alpha_{2j} - \alpha_{2j}^*)\beta_{1i}((x_i \cdot x_j) + 1) \\ & + \frac{1}{2} \sum_{i=1}^p \sum_{j=1}^p \beta_{1i}\beta_{1j}((x_i \cdot x_j) + 1) - \sum_{j=p+1}^{p+q} (\alpha_{2j} - \alpha_{2j}^*) \\ \text{s.t. } & r_3 - \sum_{j=p+1}^{p+q} \alpha_{2j}^* \geq 0, \\ & -r_4 + \sum_{i=1}^p \beta_{1i} \geq 0, \\ & 0 \leq \alpha_{2j}^*, \alpha_{2j} \leq C_3, \quad j = p+1, p+2, \dots, p+q, \\ & 0 \leq \beta_{1i} \leq C_4, \quad i = 1, 2, \dots, p.\end{aligned} \quad (32)$$

(32) can be converted into the following problem:

$$\begin{aligned}\min_{\hat{\pi}} & \frac{1}{2} \hat{\pi}^T \hat{\Lambda} \hat{\pi} + \hat{\kappa}^T \hat{\pi} \\ \text{s.t. } & r_3 - \hat{\psi}^T \hat{\pi} \geq 0, \\ & -r_4 + \hat{\delta}^T \hat{\pi} \geq 0, \\ & 0 \leq \hat{\pi} \leq \hat{C},\end{aligned} \quad (33)$$

where

$$\begin{aligned}\hat{\pi} &= (\alpha_{2j}^*, \alpha_{2j}, \beta_{1i})^T, \\ \hat{\kappa} &= (e_2^T, -e_2^T, 0e_1^T)^T, \\ \hat{\psi} &= (e_2^T, 0e_2^T, 0e_1^T)^T, \\ \hat{\delta} &= (0e_2^T, 0e_2^T, e_1^T)^T, \\ \hat{C} &= (C_3e_2^T, C_3e_2^T, C_4e_1^T)^T,\end{aligned}$$

and

$$\begin{aligned}\hat{\Lambda} &= \begin{pmatrix} \hat{\Lambda}_1 & \hat{\Lambda}_2 \\ \hat{\Lambda}_2^T & \hat{\Lambda}_3 \end{pmatrix}, \\ \hat{\Lambda}_1 &= \begin{pmatrix} BB^T + e_2 e_2^T & -BB^T - e_2 e_2^T \\ -BB^T - e_2 e_2^T & BB^T + e_2 e_2^T \end{pmatrix}, \\ \hat{\Lambda}_2 &= \begin{pmatrix} BA^T + e_2 e_1^T \\ -BA^T - e_2 e_1^T \end{pmatrix}, \\ \hat{\Lambda}_3 &= AA^T + e_1 e_1^T.\end{aligned}$$

We can find (ω_1, b_1) and (ω_2, b_2) by solving dual problems (28) and (32). A new point $x \in R^n$ is assigned to the class by the following decision function:

$$Class = \arg \min \left(\frac{|\omega_1 \cdot x + b_1 - 1|}{\|\omega_1\|}, \frac{|\omega_2 \cdot x + b_2 + 1|}{\|\omega_2\|} \right), \quad (34)$$

where $|\cdot|$ is an absolute value.

3.2. Nonlinear IMNSVM

In this section, we present the nonlinear IMNSVM. As with SVMs, since only inner products appear in dual problems (28) and (32), the kernel function $K(x, x')$ can be directly applied in these problems. In this way, the linear IMNSVM can be extended to a nonlinear classifier. When the linear kernel function is used, the nonlinear classifier can be converted into a linear classifier.

The nonlinear dual problem corresponding to problem (28) is obtained via the following steps:

$$\begin{aligned}\min_{\alpha_1^*, \alpha_1, \beta_2} \quad & \frac{1}{2} \sum_{i=1}^p \sum_{j=1}^p (\alpha_{1i} - \alpha_{1i}^*) (\alpha_{1j} - \alpha_{1j}^*) (K(x_i \cdot x_j) + 1) \\ & - \sum_{i=1}^p \sum_{j=p+1}^{p+q} (\alpha_{1i} - \alpha_{1i}^*) \beta_{2j} (K(x_i \cdot x_j) + 1) \\ & + \frac{1}{2} \sum_{i=p+1}^{p+q} \sum_{j=p+1}^{p+q} \beta_{2i} \beta_{2j} (K(x_i \cdot x_j) + 1) + \sum_{i=1}^p (\alpha_{1i}^* - \alpha_{1i}) \\ \text{s.t.} \quad & r_1 - \sum_{i=1}^p \alpha_{1i}^* \geq 0, \\ & -r_2 + \sum_{j=p+1}^{p+q} \beta_{2j} \geq 0, \\ & 0 \leq \alpha_{1i}^*, \alpha_{1i} \leq C_1, \quad i = 1, 2, \dots, p, \\ & 0 \leq \beta_{2j} \leq C_2, \quad j = p+1, p+2, \dots, p+q.\end{aligned} \quad (35)$$

(35) can be converted into the following problem:

$$\begin{aligned}\min_{\pi} \quad & \frac{1}{2} \pi^T \bar{\Lambda} \pi + \kappa^T \pi \\ \text{s.t.} \quad & r_1 - \psi^T \pi \geq 0, \\ & -r_2 + \delta^T \pi \geq 0, \\ & 0 \leq \pi \leq C,\end{aligned} \quad (36)$$

where

$$\begin{aligned}\pi &= (\alpha_1^{*T}, \alpha_1^T, \beta_2^T)^T, \\ \kappa &= (e_1^T, -e_1^T, 0e_2^T)^T, \\ \psi &= (e_1^T, 0e_1^T, 0e_2^T)^T, \\ \delta &= (0e_1^T, 0e_1^T, e_2^T)^T,\end{aligned}$$

$$C = (C_1 e_1^T, C_1 e_1^T, C_2 e_2^T)^T,$$

and

$$\begin{aligned}\bar{\Lambda} &= \begin{pmatrix} \bar{\Lambda}_1 & \bar{\Lambda}_2 \\ \bar{\Lambda}_2^T & \bar{\Lambda}_3 \end{pmatrix}, \\ \bar{\Lambda}_1 &= \begin{pmatrix} K(A, A^T) + e_1 e_1^T & -K(A, A^T) - e_1 e_1^T \\ -K(A, A^T) - e_1 e_1^T & K(A, A^T) + e_1 e_1^T \end{pmatrix}, \\ \bar{\Lambda}_2 &= \begin{pmatrix} K(A, B^T) + e_1 e_2^T \\ -K(A, B^T) - e_1 e_2^T \end{pmatrix}, \\ \bar{\Lambda}_3 &= K(B, B^T) + e_2 e_2^T.\end{aligned}$$

Similarly, we have the nonlinear dual problem corresponding to problem (32):

$$\begin{aligned}\min_{\alpha_2^*, \alpha_2, \beta_1} \quad & \frac{1}{2} \sum_{i=p+1}^{p+q} \sum_{j=p+1}^{p+q} (\alpha_{2i} - \alpha_{2i}^*)(\alpha_{2j} - \alpha_{2j}^*)(K(x_i \cdot x_j) + 1) \\ & - \sum_{i=1}^p \sum_{j=p+1}^{p+q} (\alpha_{2j} - \alpha_{2j}^*) \beta_{1i} (K(x_i \cdot x_j) + 1) \\ & + \frac{1}{2} \sum_{i=1}^p \sum_{j=1}^p \beta_{1i} \beta_{1j} (K(x_i \cdot x_j) + 1) - \sum_{j=p+1}^{p+q} (\alpha_{2j} - \alpha_{2j}^*) \\ \text{s.t.} \quad & r_3 - \sum_{j=p+1}^{p+q} \alpha_{2j}^* \geq 0, \\ & -r_4 + \sum_{i=1}^p \beta_{1i} \geq 0, \\ & 0 \leq \alpha_{2j}^*, \alpha_{2j} \leq C_3, \quad j = p+1, p+2, \dots, p+q, \\ & 0 \leq \beta_{1i} \leq C_4, \quad i = 1, 2, \dots, p.\end{aligned} \tag{37}$$

(37) can be converted into the following problem:

$$\begin{aligned}\min_{\hat{\pi}} \quad & \frac{1}{2} \hat{\pi}^T \tilde{\Lambda} \hat{\pi} + \hat{\kappa}^T \hat{\pi} \\ \text{s.t.} \quad & r_3 - \hat{\psi}^T \hat{\pi} \geq 0, \\ & -r_4 + \hat{\delta}^T \hat{\pi} \geq 0, \\ & 0 \leq \hat{\pi} \leq \hat{C},\end{aligned} \tag{38}$$

where

$$\begin{aligned}\hat{\pi} &= (\alpha_2^{*T}, \alpha_2^T, \beta_1^T)^T, \\ \hat{\kappa} &= (e_2^T, -e_2^T, 0e_1^T)^T, \\ \hat{\psi} &= (e_2^T, 0e_2^T, 0e_1^T)^T, \\ \hat{\delta} &= (0e_2^T, 0e_2^T, e_1^T)^T, \\ \hat{C} &= (C_3 e_2^T, C_3 e_2^T, C_4 e_1^T)^T,\end{aligned}$$

and

$$\begin{aligned}\tilde{\Lambda} &= \begin{pmatrix} \tilde{\Lambda}_1 & \tilde{\Lambda}_2 \\ \tilde{\Lambda}_2^T & \tilde{\Lambda}_3 \end{pmatrix}, \\ \tilde{\Lambda}_1 &= \begin{pmatrix} K(B, B^T) + e_2 e_2^T & -K(B, B^T) - e_2 e_2^T \\ -K(B, B^T) - e_2 e_2^T & K(B, B^T) + e_2 e_2^T \end{pmatrix}, \\ \tilde{\Lambda}_2 &= \begin{pmatrix} K(B, A^T) + e_2 e_1^T \\ -K(B, B^T) - e_2 e_1^T \end{pmatrix}, \\ \tilde{\Lambda}_3 &= K(A, A^T) + e_1 e_1^T.\end{aligned}$$

4. Experimental Results

In this section, the feasibility and effectiveness of the IMNSVM are explained through numerical experiments utilizing Algorithm 1. The experimental data are all from UCI open databases [29]. The existing models C-SVM, TBSVM, NPSVM, and NSVM are compared with the IMNSVM using different data sets. All the experimental methods were employed using MATLAB 2018b on a PC with an Intel Core I5 processor and 8 GB of RAM. All algorithms were executed using QP in MATLAB Toolbox. Before training, all data points were normalized to ensure that their features were between [0, 1]. For multi-classification problems, we adopted the One-vs.-Rest (OvR) strategy.

Algorithm 1 (IMNSVM)

- (1) Input the training set (1).
- (2) Select the appropriate parameters $C_i > 0, r_i > 0, i = 1, 2, 3, 4$ and kernel function $K(x, x')$.
- (3) By solving two QPPs (35) and (37), the optimal solutions $(\alpha_1^{*T}, \alpha_1^T, \beta_2^T)^T$ and $(\alpha_2^{*T}, \alpha_2^T, \beta_1^T)^T$ are obtained.
- (4) For a new point $x \in R^n$, assign it using

$$Class = \arg \min_{k=1,2} \frac{|f_k(x)|}{\|\Delta_k\|},$$

where

$$f_1(x) = \sum_{i=1}^p (\alpha_{1i} - \alpha_{1i}^*) K(x_i, x) - \sum_{j=p+1}^{p+q} \beta_{2j} K(x_j, x) + b_1 - 1,$$

$$f_2(x) = - \sum_{j=p+1}^{p+q} (\alpha_{2j} - \alpha_{2j}^*) K(x_j, x) + \sum_{i=1}^p \beta_{1i} K(x_i, x) + b_2 + 1,$$

$$\Delta_1 = \pi^T \bar{\Lambda} \pi, \quad \Delta_2 = \hat{\pi}^T \tilde{\Lambda} \hat{\pi},$$

and b_1, b_2 are computed by (23) and (31).

We adopted the ten-fold cross-validation method, which can better determine the classification of imbalanced data sets. Each data set is randomly divided into ten parts. Nine of them are used as the training set, and the rest are used as the test set. This process is repeated ten times, and the average of the ten results is taken to be the final classification accuracy. The parameters $C_i, i = 1, 2, 3, 4$ are tuned to optimize classification accuracy in the range $[2^{-5}, 2^7]$. To reduce the time taken to select parameters, we suppose that $C_1 = C_3, C_2 = C_4$.

In the linear case, each of the above methods was tested. The results obtained using the methods above are shown in Table 1. Those marked in bold denote the optimal accuracy.

In the nonlinear case, the RBF kernel $K(x, x') = \exp(\frac{-\|x-x'\|^2}{\sigma})$ is applied, where σ is tuned for optimal classification accuracy in the range $[2^{-8}, 2^8]$. Table 2 shows the results regarding the nonlinear classifiers with respect to the benchmark data sets.

From Tables 1 and 2, we can see that the IMNSVM has better classification performance on more than half of the data sets in the linear and nonlinear cases. But it can also be seen that the IMNSVM is not applicable to certain data sets. We will determine and analyze the reasons for this using more data sets in the future. Of course, classification accuracy cannot fully demonstrate the advantages of this algorithm. We first studied the effectiveness of the algorithm; then, we conducted comparative studies on other performance measures, such as computation time, sparseness, and computational complexity. If $\rho = 1$, the IMNSVM approximates the NSVM. From the calculation results, ρ , as a variable, can better regulate the margin between two classes of points and improve classification accuracy. The variable ρ may not be very large, and it may even be less than 1.

Table 1. Results of linear classifiers with respect to the benchmark data sets.

Data	C-SVM Accuracy%	TBSVM Accuracy%	NPSVM Accuracy%	NSVM Accuracy%	IMNSVM Accuracy %
Spect (267 × 44)	75.32 ± 3.07	72.64 ± 3.56	79.42 ± 1.52	79.40 ± 0.16	79.59 ± 0.18
CMC (1473 × 9)	70.31 ± 4.38	67.28 ± 2.21	74.03 ± 2.16	76.32 ± 4.47	77.39 ± 0.12
German (1000 × 20)	71.96 ± 3.17	73.18 ± 2.37	74.96 ± 3.35	73.51 ± 1.93	70.00 ± 0.00
Hepatitis (155 × 19)	82.98 ± 4.16	80.79 ± 2.24	85.19 ± 5.51	86.27 ± 3.18	88.74 ± 2.05
Haberman (306 × 3)	72.65 ± 2.03	73.47 ± 3.26	72.36 ± 4.70	72.58 ± 3.04	73.53 ± 0.47
WPBC (198 × 34)	73.25 ± 4.38	74.19 ± 3.25	75.29 ± 4.40	78.80 ± 4.28	79.85 ± 4.86
Sonar (208 × 60)	80.59 ± 4.57	80.92 ± 4.51	79.98 ± 3.10	81.62 ± 3.25	79.37 ± 1.34
BTSC (748 × 5)	76.89 ± 0.79	77.14 ± 1.60	76.21 ± 0.41	76.60 ± 0.52	77.52 ± 0.24
Parkinsons (195 × 22)	73.19 ± 2.99	77.21 ± 11.51	82.32 ± 4.54	83.47 ± 6.03	79.39 ± 1.97
Breast (683 × 9)	96.03 ± 2.09	96.49 ± 1.86	93.98 ± 3.99	97.05 ± 0.64	97.86 ± 1.26

Table 2. Results of nonlinear classifiers on the benchmark data sets.

Data	C-SVM Accuracy%	TBSVM Accuracy%	NPSVM Accuracy%	NSVM Accuracy%	IMNSVM Accuracy%
Spect (267 × 44)	75.44 ± 3.60	77.60 ± 2.11	79.39 ± 0.19	79.26 ± 1.75	80.15 ± 2.23
CMC (1473 × 9)	70.39 ± 4.26	72.63 ± 3.12	75.12 ± 2.65	77.24 ± 1.17	77.55 ± 0.32
German (1000 × 20)	74.25 ± 1.49	72.13 ± 2.72	74.71 ± 3.13	76.55 ± 4.49	70.30 ± 0.51
Hepatitis (155 × 19)	81.17 ± 4.62	84.37 ± 4.52	85.90 ± 4.01	85.65 ± 4.24	86.67 ± 3.83
Haberman (306 × 3)	72.65 ± 2.03	73.21 ± 2.27	73.43 ± 0.48	73.86 ± 2.19	73.93 ± 1.45
WPBC (198 × 34)	75.36 ± 4.73	74.27 ± 1.06	75.35 ± 5.68	76.27 ± 2.03	76.28 ± 0.81
Sonar (208 × 60)	83.40 ± 4.57	84.30 ± 5.03	90.86 ± 5.35	91.36 ± 6.80	89.56 ± 2.67
BTSC (748 × 5)	74.73 ± 3.43	73.28 ± 3.10	74.11 ± 2.90	75.43 ± 1.32	76.34 ± 0.28
Parkinsons (195 × 22)	71.19 ± 3.99	78.31 ± 5.51	81.45 ± 3.64	83.07 ± 3.03	76.92 ± 1.97
Breast (683 × 9)	92.13 ± 1.08	91.49 ± 1.36	93.03 ± 1.09	92.05 ± 0.84	94.72 ± 1.69

Those marked in bold denote the optimal accuracy.

5. Conclusions

In this paper, a new binary classifier, the IMNSVM, is proposed. The IMNSVM introduces the one-sided ε -band to better fit the dense regions of the training points and endow the model with sparseness. Furthermore, by introducing a new variable, ρ , the IMNSVM keeps each class of training points at a certain distance from the hyperplane

corresponding to another class of training points, keeping them as far away as possible to better separate different classes of the training points. The IMNSVM only needs to solve two small quadratic programming problems and does not require computation of the inverse of the matrix. In the nonlinear case, the kernel functions can be directly introduced. This new binary classifier, the IMNSVM, is a nonparallel support vector machine, but it can degenerate into the standard SVM under certain conditions. So, it can achieve conversion between parallel and nonparallel algorithms and is convenient for the design and comparison of the algorithms. In the future, we hope to consider similar extension algorithms for other data-mining problems such as regression, semi-supervised, and unsupervised problems.

Author Contributions: S.L. proposed the topic, participated in drafting the first manuscript, and made the final revisions. J.Y. was the lead author of the first draft. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Natural Science Foundation of Shandong Province (ZR2025MS41 and ZR2025MS86).

Data Availability Statement: The data sets were derived from the UCI repository of machine learning databases: <https://archive.ics.uci.edu/> (accessed on 1 January 2026).

Acknowledgments: The authors thank all the anonymous reviewers for their helpful comments.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Cortes, C.; Vapnik, V.N. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
2. Vapnik, V.N. *The Nature of Statistical Learning Theory*; Springer: New York, NY, USA, 1996.
3. Vapnik, V.N. *Statistical Learning Theory*; John Wiley and Sons: New York, NY, USA, 1998.
4. Liu, Y.X.; Peng, Y.H.; Zhou, X.H. Texture classification using finite ridgelet transform and support vector machines. In *PRICAI 2006: Trends in Artificial Intelligence*; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2006; Volume 4099, pp. 1042–1046.
5. Isa, D.; Lee, L.H.; Kallimani, V.P.; RajKumar, R. Text document preprocessing with the Bayes formula for classification using the support vector machine. *IEEE Trans. Knowl. Data Eng.* **2008**, *20*, 1264–1272. [[CrossRef](#)]
6. Wu, Y.C.; Lee, Y.S.; Yang, J.C. Robust and efficient multiclass SVM models for phrase pattern recognition. *Pattern Recognit.* **2008**, *41*, 2874–2889. [[CrossRef](#)]
7. Wang, X.Y.; Wang, T.; Bu, J.A. Color image segmentation using pixel wise support vector machine classification. *Pattern Recognit.* **2011**, *44*, 777–787. [[CrossRef](#)]
8. Khan, N.M.; Ksantini, R.; Ahmad, I.S.; Boufama, B. A novel SVM+ NDA model for classification with an application to face recognition. *Pattern Recognit.* **2012**, *45*, 66–79. [[CrossRef](#)]
9. Gong, R.F.; Chu, M.X.; Yang, Y.H.; Feng, Y. A multi-class classifine based on support vector hyper-spheres for steel plate surface defects. *Chemom. Intell. Lab. Syst.* **2019**, *188*, 70–78. [[CrossRef](#)]
10. Cao, X.H.; Wang, D.; Wang, X.Z.; Zhao, J.; Jiao, L.C. Hyperspectral imagery classification with cascaded support vector machines and multi-scale superpixel segmentation. *Int. J. Remote Sens.* **2020**, *41*, 4528–4548. [[CrossRef](#)]
11. Dhankhar, S.; Gupta, M.K.; Memon, F.H.; Bhatia, S.; Dadheech, P.; Mashat, A. Support vector machine based handwritten hindi character recognition and summarization. *Comput. Syst. Sci. Eng.* **2022**, *43*, 397–412. [[CrossRef](#)]
12. Lei, W.L.; Lei, Y.; Li, B.; Jia, K. Support vector machines optimisation for face recognition using sparrow search algorithm. *Trait. Signal* **2023**, *40*, 2009–2017. [[CrossRef](#)]
13. Mangasarian, O.L.; Wild, E.W. Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE Trans. Pattern Anal. Mach. Intell.* **2006**, *28*, 69–74. [[CrossRef](#)]
14. Jayadara; Khemchandani, R.; Chandra, S. Twin support vector machines for pattern classification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 905–910. [[CrossRef](#)] [[PubMed](#)]
15. Kumar, M.A.; Gopal, M. Application of smoothing technique on twin support vector machines. *Pattern Recognit. Lett.* **2008**, *29*, 1842–1848. [[CrossRef](#)]
16. Chen, X.B.; Yang, J.; Ye, Q.L.; Liang, J. Recursive projection twin support vector machine via within-class variance minimization. *Pattern Recognit.* **2011**, *44*, 2643–2655. [[CrossRef](#)]

17. Shao, Y.H.; Zhang, C.H.; Wang, X.B.; Deng, N.Y. Improvements on twin support vector machines. *IEEE Trans. Neural Netw.* **2011**, *22*, 962–968. [[CrossRef](#)]
18. Tian, Y.J.; Qi, Z.Q.; Ju, X.C.; Shi, Y.; Liu, X.H. Nonparallel support vector machines for pattern classification. *IEEE Trans. Cybern.* **2014**, *44*, 1067–1079. [[CrossRef](#)]
19. Zhao, J.; Yang, Z.J.; Xu, Y.T. Nonparallel least square support vector machine for classification. *Appl. Intell.* **2016**, *45*, 1119–1128. [[CrossRef](#)]
20. Liu, L.M.; Chu, M.X.; Gong, R.F.; Zhang, L. An improved nonparallel support vector machine. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 5129–5143. [[CrossRef](#)]
21. Sun, F.M.; Lian, S.J. v -Improved nonparallel support vector machine. *Sci. Rep.* **2022**, *12*, 17855. [[CrossRef](#)]
22. Qi, K.; Yang, H. LS-GNHSVM: A novel joint geometrical nonparallel hyperplane support vector machine. *Expert Syst. Appl.* **2023**, *215*, 119413. [[CrossRef](#)]
23. Li, C.N.; Shao, Y.H.; Wang, H.J.; Huang, L.W.; Zhao, Y.T.; Xiu, N.H.; Deng, N.Y. Union nonparallel support vector machines framework with consistency. *Appl. Soft Comput.* **2023**, *136*, 110129. [[CrossRef](#)]
24. Gao, R.Y.; Qi, K.; Yang, H. Fused robust geometric nonparallel hyperplane support vector machine for pattern classification. *Expert Syst. Appl.* **2024**, *236*, 121331. [[CrossRef](#)]
25. Shao, W.Z.; Shao, Y.H.; Li, C.N. A novel regularization method for decorrelation learning of non-parallel hyperplanes. *Inf. Sci.* **2024**, *667*, 120461. [[CrossRef](#)]
26. Wang, H.M.; Li, P.; Zheng, Y.Y.; Jiang, K.; Xu, Y.T. Sparse pinball Universum nonparallel support vector machine and its safe screening rule. *Appl. Intell.* **2025**, *55*, 563. [[CrossRef](#)]
27. Shao, Y.H.; Yang, K.L.; Liu, M.Z.; Wang, Z.; Li, C.N.; Chen, W.J. From support vector machine to nonparallel support vector machine. *Oper. Res. Trans.* **2018**, *22*, 55–65.
28. Mangasarian, O.L. *Nonlinear Programming*; SIAM: Philadelphia, PA, USA, 1994.
29. Blake, C.L.; Merz, C.J. *UCI Repository of Machine Learning Databases*; Department of Computer Science, University of California: Irvine, CA, USA, 1998. Available online: <https://archive.ics.uci.edu/> (accessed on 1 January 2026).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.