[1] **A. Putthakun**

[2] **T. Mouktonglang**

Hybrid Optimization Techniques for Solving Twin Support Vector Machines Using ADMM and IPM

Journal of Electrical Systems

Abstract: - In this research, we explore the enhancement of a Twin Support Vector Machine (TWSVM) by combining the Alternating Direction Method of Multipliers (ADMM) and the Interior Point Method (IPM). TWSVM is a binary classification method that constructs two non-parallel hyperplanes by solving a pair of quadratic programming problems. To address the sensitivity of TWSVM to noise and outliers, we extend the formulation to a robust variant, incorporating regularization techniques to improve performance in noisy environments, termed the Robust Twin Support Vector Machine (RTWSVM). ADMM is employed to decompose the optimization problem into smaller subproblems, offering computational efficiency for large-scale datasets. Meanwhile, IPM is utilized for its globally convergent properties, ensuring precise optimization even in the presence of complex constraints. In iteration of ADMM, the subproblem involving argmin optimization is solved efficiently using IPM, ensuring both accuracy and rapid convergence. This hybrid approach combines the scalability of ADMM with the optimization precision of IPM. Experimental results demonstrate the computational efficiency and robustness of the proposed method, particularly on large-scale, noisy datasets. The method is evaluated on real-world datasets such as Breast Cancer Wisconsin, Dry Bean, Heart Disease, and Pima Indians Diabetes, offering valuable insights into its practical applications.

Keywords: - Optimization, binary classification, twin support vector machines (TWSVM), the alternating direction method of multipliers (ADMM), interior Point Method (IPM)

I. INTRODUCTION

Support Vector Machines (SVM) are widely used for binary classification tasks due to their ability to maximize the margin between classes, making them effective in many real-world applications [1][5]. However, traditional SVM methods often face challenges in handling large-scale datasets, noisy environments and imbalanced data distributions [6]. These limitations restrict their practicality, especially when applied to complex datasets with high dimensionality or significant noise.

Twin Support Vector Machines (TWSVM) was introduced to address some of these challenges by constructing two non-parallel hyperplanes to classify data more efficiently. Compared to traditional SVM, TWSVM reduces computational complexity and improves efficiency [6]. However, despite these advantages, TWSVM remains sensitive to noise and outliers, leading to reduced robustness and classification accuracy in real-world scenarios.

This research aims to overcome these limitations by proposing a hybrid optimization framework combining the strengths of the Alternating Direction Method of Multipliers (ADMM) and the Interior Point Method (IPM). ADMM provides computational efficiency by decomposing the optimization problem into smaller, more manageable subproblems, while IPM ensures precise optimization under complex constraints [3], [4]. The hybrid approach leverages the scalability of ADMM and the optimization accuracy of IPM, making it well-suited for large-scale and noisy datasets.

Additionally, this study introduces a novel adjustment to the TWSVM margin constraints by incorporating an adaptive parameter, $d(1-\alpha)$ which balances margin expansion and misclassification tolerance. This adjustment enhances the model's robustness and generalization performance, particularly in challenging scenarios involving noise or class imbalance [7].

The contributions of this work include the development of a robust hybrid optimization method and the modification of TWSVM margin constraints to improve performance in real-world classification tasks. The proposed method provides a significant step forward in addressing the practical limitations of existing classification algorithms.

The rest of this paper is organized as follows: Section II reviews related work on SVM, TWSVM, ADMM, and IPM. Section III details the proposed hybrid optimization method and the robust TWSVM formulation. Section IV presents experimental results and analysis, while Section V concludes the study with key findings and potential directions for future research.

[1] Department of Mathematics, Faculty of Science, Chiang Mai University, Chiang Mai 50200, Thailand. anawinputtakun@gmail.com
[2] Department of Mathematics, Faculty of Science, Chiang Mai University, Chiang Mai 50200, Thailand. thanasak.m@cmu.ac.th
* Corresponding Author Email: hamsalekha20@gmail.com

II. BACKGROUND

This section reviews foundational concepts essential for understanding the proposed hybrid optimization framework. It provides a detailed overview of Support Vector Machines (SVM), Twin Support Vector Machines (TWSVM), the Alternating Direction Method of Multipliers (ADMM), and the Interior Point Method (IPM). These methods form the core of this research, addressing various challenges in classification tasks and optimization.

2.1 Support Vector Machine (*SVM*)

SVM are supervised learning models with associated learning algorithms that analyzes data for classification and regression. The idea of SVM is to find an optimal hyperplane that separates different classes with the largest possible margin.

The SVM optimization problem is formulated to minimize the following objective function:

$$\begin{aligned} &\min \ \frac{1}{2}\|w\|_2^2, \\ &s.t. \quad y_i(wx_i - b) \geq 1, \forall i, \end{aligned} \tag{1}$$

where w is the weight vector, b is the bias term and x_i , y_i are the training data points and their labels.

2.2 Twin Support Vector Machine (*TWSVM*)

The idea of TWSVM is to find two hyperplanes in which samples are assigned to a class according to their distance from the hyperplanes and each of them is the closest to the samples of its own class and the farthest from the samples of the opposite class. Equations of the two hyperplanes are as follows,

$$w_1^T x + b_1 = 0 \ , \ w_2^T x + b_2 = 0. \tag{2}$$

Assume a binary classification task with classes $+1$ and -1. A and B are matrices of samples belonging to classes $+1$ and -1, respectively. The two hyperplanes of TWSVM obtained by solving the following objective functions.

$$\begin{aligned} &\min_{w_1,b_1,\xi_2} \ \frac{1}{2}\|Aw_1 + e_1 b_1\|_2^2 + \frac{c_1}{2}\|\xi_2\|_1, \\ &s.t. \quad -\left(Bw_1 + e_2 b_1\right) + \xi_2 \geq e_2, \ \ \xi_2 \geq 0, \end{aligned} \tag{3}$$

$$\begin{aligned} &\min_{w_2,b_2,\xi_1} \ \frac{1}{2}\|Bw_2 + e_2 b_2\|_2^2 + \frac{c_2}{2}\|\xi_1\|_1, \\ &s.t. \quad \left(Aw_2 + e_1 b_2\right) + \xi_1 \geq e_1, \ \ \xi_1 \geq 0, \end{aligned} \tag{4}$$

where e is a column vector of ones in real space of arbitrary dimension and ξ_i represents the vector of slack variables of size n .

2.3 Alternating Direction Method of Multipliers (*ADMM*)

ADMM is an optimization algorithm that handles complex, large-scale problems by decomposing them into smaller subproblems. It is particularly useful for constrained optimization problems, often applied in machine learning and support vector machines. ADMM minimizes a constrained objective function of the following form:

$$\begin{aligned} &\min_{x,z} \ \ f(x) + g(z), \\ &s.t. \quad Ax + Bz = c, \end{aligned} \tag{5}$$

by using an Augmented Lagrangian approach:

$$L_\rho(x, y, z) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\rho}{2}\|Ax + Bz - c\|_2^2, \tag{6}$$

where y is the Lagrange multiplier and ρ is a penalty parameter.

ADMM iterative process includes:

1. Update x by minimizing L_ρ with respect to x , keeping z and y fixed.
2. Update z by minimizing L_ρ with respect to z , keeping x and y fixed.

3. Update y by adjusting y to better satisfy the constraint $Ax + Bz = c$.

2.4 Interior Point Method (*IPM*)

The Interior Point Method (IPM) is an efficient optimization technique for solving constrained optimization problems, particularly those involving convex functions. It is widely used in machine learning for solving quadratic programming (QP) problems.

Consider a general optimization problem:

$$\min_{x} \quad f(x),$$
$$s.t. \quad g(x) \le 0,\ x \ge 0, \tag{7}$$

where $f(x)$ is the objective function, $g(x) \le 0$ represents inequality constraints and $x \ge 0$ represents non-negative constraints.

IPM solves such problems by introducing a log-barrier function for the inequality constraints:

$$\phi(x;\mu) = f(x) - \mu_1 \sum_{i=1}^{m} \ln(-g_i(x)) - \mu_2 \sum_{j=1}^{n} \ln(x),\ \ \mu_1, \mu_2 > 0, \tag{8}$$

where $\mu > 0$ is the barrier parameter. The algorithm solves a sequence of approximations to this problem, gradually reducing μ to approach the true solution.

IPM algorithm proceed step as follow:

1. initialize: Choose a feasible starting point x_0, initial barrier parameter μ_0 and tolerance ε.
2. iterative: For each μ:
   - Solve the barrier problem using Newton's method.
   - Update $\mu : \mu \leftarrow \beta\mu$ where $\beta \in (0,1)$.
3. Convergence: Stop when $\left\|\nabla \phi(x;\mu)\right\| < \varepsilon$.

III. PROPOSED METHOD

This section describes the methodology used to solve the TWSVM problem, by combining the ADMM and IPM. The standard TWSVM formulation is extended to a robust variant, Robust Twin Support Vector Machine (RTWSVM), to improve performance in noisy environments. ADMM is used to decompose the optimization problem into smaller subproblems, while IPM ensures efficient and accurate optimization during each iteration. The following subsections detail the formulation of the problem, the hybrid approach, and the computational procedures.

Consider the TWSVM optimization problem

$$\min_{w_1, b_1, \xi_2} \ \frac{1}{2}\left\|Aw_1 + e_1 b_1\right\|_2^2 + \frac{c_1}{2}\left\|\xi_2\right\|_1,$$
$$s.t. \quad -\left(Bw_1 + e_2 b_1\right) + \xi_2 \ge e_2,\ \ \xi_2 \ge 0,$$
$$\min_{w_2, b_2, \xi_1} \ \frac{1}{2}\left\|Bw_2 + e_2 b_2\right\|_2^2 + \frac{c_2}{2}\left\|\xi_1\right\|_1,$$
$$s.t. \quad \left(Aw_2 + e_1 b_2\right) + \xi_1 \ge e_1,\ \ \xi_1 \ge 0,$$

for simplicity, let $F = \begin{bmatrix} A & e_1 \end{bmatrix}$, $E = \begin{bmatrix} B & e_2 \end{bmatrix}$, $x_i = \begin{bmatrix} w_i \\ b_i \end{bmatrix}$, $p_i = \frac{c_i}{2}$ where $i = 1, 2$ and $z_i \ge 0$ is slack variables.

Then, TWSVM will be in form of:

$$\min_{x_1, \xi_2, z_1} \ \frac{1}{2}\left\|Fx_1\right\|_2^2 + p_1\left\|\xi_2\right\|_1, \tag{9}$$
$$s.t. \quad -\left(Ex_1\right) + \xi_2 - z_1 = e_2,\ \ \xi_2 \ge 0,$$

$$\min_{w_2,b_2,\xi_1} \ \frac{1}{2}\|Ex_2\|_2^2 + p_2\|\xi_1\|_1, \tag{10}$$
$$s.t. \quad Fx_2 + \xi_1 - z_2 = e_1, \ \xi_1 \geq 0.$$

By using augmented Lagrangian function to (9) and (10), we get the following problem:

$$L_\rho(x_1,\xi_2,z_1,y_1) = \frac{1}{2}\|Fx_1\|_2^2 + p_1\|\xi_2\|_1 + y^T(Ex_1 - \xi_2 + z_1 + e_2) + \frac{\rho}{2}\|Ex_1 - \xi_2 + z_1 + e_2\|_2^2, \tag{11}$$

$$L_\rho(x_2,\xi_1,z_2,y_2) = \frac{1}{2}\|Ex_2\|_2^2 + p_2\|\xi_1\|_1 + y^T(Fx_2 + \xi_1 - z_2 - e_1) + \frac{\rho}{2}\|Fx_2 + \xi_1 - z_2 - e_1\|_2^2, \tag{12}$$

where y_1, y_2 are Lagrangian multiplier and ρ is parameter.

Then, ADMM consists of iterations for (11) as follows:

$$x_1^{k+1} = \arg\min_{x_1}\left(L_\rho\left(x_1,\xi_2^k,z_1^k,y_1^k\right)\right), \tag{13}$$

$$\xi_2^{k+1} = \arg\min_{\xi_2}\left(L_\rho\left(x_1^{k+1},\xi_2,z_1^k,y_1^k\right)\right), \ \xi_2 \geq 0, \tag{14}$$

$$z_1^{k+1} = \arg\min_{z_1}\left(L_\rho\left(x_1^{k+1},\xi_2^{k+1},z_1,y_1^k\right)\right), \ z_1 \geq 0, \tag{15}$$

$$y_1^{k+1} = y_1^k + Ex_1^{k+1} - \xi_2^{k+1} + z_1^{k+1} + e_2. \tag{16}$$

And, ADMM consists of iterations for (12) as follows:

$$x_2^{k+1} = \arg\min_{x_2}\left(L_\rho\left(x_2,\xi_1^k,z_2^k,y_2^k\right)\right), \tag{17}$$

$$\xi_1^{k+1} = \arg\min_{\xi_1}\left(L_\rho\left(x_2^{k+1},\xi_1,z_2^k,y_2^k\right)\right), \ \xi_1 \geq 0, \tag{18}$$

$$z_2^{k+1} = \arg\min_{z_2}\left(L_\rho\left(x_2^{k+1},\xi_1^{k+1},z_2,y_2^k\right)\right), z_2 \geq 0, \tag{19}$$

$$y_2^{k+1} = y_2^k + Fx_2^{k+1} + \xi_1^{k+1} - z_2^{k+1} - e_1. \tag{20}$$

3.1 ADMM with IPM on TWSVM

In this part, we address the optimization subproblems that arise during the ADMM iterations for both hyperplanes in TWSVM. Specifically, we focus on solving all subproblems in the ADMM framework, with particular emphasis on the subproblem for x^{k+1}, which will be handled using the Interior Point Method (IPM).

Consider (13)

$$x_1^{k+1} = \arg\min_{x_1}\left(L_\rho\left(x_1,\xi_2^k,z_1^k,y_1^k\right)\right),$$
$$= \arg\min_{x_1}\left(\frac{1}{2}\left(x_1^T F^T F x_1\right) + \left(y_1^k\right)^T\left(Ex_1\right) + \frac{\rho}{2}\left\|Ex_1 - \xi_2^k + z_1^k + e_2\right\|_2^2\right).$$

We apply the Interior Point Method (IPM) to efficiently solve this argmin problem, leveraging its capability to handle constrained optimization with high precision. Specifically, we utilize the quadprog solver in MATLAB, which includes the IPM as one of its optimization algorithms, to solve quadratic programming problems within the ADMM framework.

Next, ξ_2^{k+1} is solved by Proximal operators.

Consider (14)

$$\xi_2^{k+1} = \arg\min_{\xi_2}\left(L_\rho\left(x_1^{k+1}, \xi_2, z_1^k, y_1^k\right)\right),\ \xi_2 \geq 0$$

$$= \arg\min_{\xi_2}\left(p_1\|\xi_2\|_1 - \left(y_1^k\right)^T \xi_2 + \frac{\rho}{2}\left\|Ex_1^{k+1} - \xi_2 + z_1^k + e_2\right\|_2^2\right),\ \xi_2 \geq 0,$$

$$= \arg\min_{\xi_2}\left(\|\xi_2\|_1 + \frac{\rho}{2p_1}\left\|Ex_1^{k+1} - \xi_2 + z_1^k + e_2 - \frac{y_1^k}{\rho}\right\|_2^2\right),\ \xi_2 \geq 0,$$

$$= \arg\min_{\xi_2}\left(\|\xi_2\|_1 + \frac{1}{2\frac{p_1}{\rho}}\left\|\xi_2 - \left(Ex_1^{k+1} + z_1^k + e_2 - \frac{y_1^k}{\rho}\right\|_2^2\right),\ \xi_2 \geq 0.$$

By using Proximal operator

$$\text{Prox}_f(v) = \arg\min_s\left(f(s) + \frac{1}{2\alpha}\|s - v\|_2^2\right) \equiv s_\alpha(v),$$

where the *soft thresholding operator* $s_\alpha(v)$ is defined as

$$s_\alpha(v) = \begin{cases} v - \alpha & , v > a, \\ 0 & , |v| \leq \alpha, \\ v + \alpha & , v < -\alpha. \end{cases}$$

Then, we get the form of ξ_2^{k+1} as follow:

$$\xi_2^{k+1} = \xi_{2\frac{p_1}{\rho}}(v) = \begin{cases} v - \frac{p_1}{\rho} & , v > \frac{p_1}{\rho}, \\ 0 & , |v| \leq \frac{p_1}{\rho}, \\ v + \frac{p_1}{\rho} & , v < -\frac{p_1}{\rho}, \end{cases}$$

where $v = Ex_1^{k+1} + z_1^k + e_2 + \frac{y_1^k}{\rho}$.

Next, z_1^{k+1} is solved by differentiating with respect to its parameters.

Consider (15)

$$z_1^{k+1} = \arg\min_{z_1}\left(L_\rho\left(x_1^{k+1}, \xi_2^{k+1}, z_1, y_1^k\right)\right),\ z_1 \geq 0,$$

$$= \arg\min_{z_1}\left(y_1^k z_1 + \frac{\rho}{2}\left\|Ex_1^{k+1} - \xi_2^{k+1} + z_1^k + e_2\right\|_2^2\right), z_1 \geq 0,$$

$$\frac{\partial L_\rho\left(x_1^{k+1}, \xi_2^{k+1}, z_1, y_1^k\right)}{\partial z_1} = 0, \Rightarrow\ y_1^k + \rho\left(Ex_1^{k+1} - \xi_2^{k+1} + z_1^{k+1} + e_2\right) = 0,\ z_1 \geq 0,$$

$$z_1^{k+1} = \xi_2^{k+1} - \left(Ex_1^{k+1} + e_2 + \frac{y_1^k}{\rho}\right),\ z_1 \geq 0.$$

To ensure the feasibility of the optimization constraints during ADMM iterations, z must satisfy $z \geq 0$. In cases where any component of z violates this condition ($z \leq 0$), a projection is applied to enforce the non-negativity constraint. This projection is defined as:

$$z = \max(0, z),$$

where any negative component of z is set to 0 .

Additionally, to maintain the balance of the optimization constraints, the removed components of z are compensated by updating the corresponding slack variable ξ, ensuring that the augmented Lagrangian formulation remains valid. The adjusted formulation can be expressed as:

$$\xi_{new} = \xi_{old} + z, \text{ if } z \le 0.$$

This projection process stabilizes the optimization procedure, prevents infeasibility, and ensures compliance with the problem's constraints.

In the last step, y_1^{k+1} is update as

$$y_1^{k+1} = y_1^k + \rho\left(Ex_1^{k+1} - \xi_2^{k+1} + z_1^{k+1} + e_2\right).$$

To simplify the process, we will let $\frac{y_1^k}{\rho} = u_1^k$.

Then, we get $u_1^{k+1} = u_1^k + Ex_1^{k+1} - \xi_2^{k+1} + z_1^{k+1} + e_2$

Thus, ADMM proceed iterations for (11) as follow:

$$x_1^{k+1} = \arg\min_{x_1}\left(\frac{1}{2}\left(x_1^T F^T F x_1\right) + \left(y_1^k\right)^T\left(Ex_1\right) + \frac{\rho}{2}\left\|Ex_1 - \xi_2^k + z_1^k + e_2\right\|_2^2\right),$$

$$\xi_2^{k+1} = \xi_{2\frac{p_1}{\rho}}(v), \text{ where } v = Ex_1^{k+1} + z_1^k + e_2 + u_1^k,$$

$$z_1^{k+1} = \xi_2^{k+1} - \left(Ex_1^{k+1} + e_2 + u_1^k\right),\ z_1 \ge 0,$$

$$u_1^{k+1} = u_1^k + Ex_1^{k+1} - \xi_2^{k+1} + z_1^{k+1} + e_2.$$

The pseudocode of the model for TWSVM 1st plane is provide in Algorithm 1.

Algorithm 1 ADMM with IPM on TWSVM 1st plane

```
Input : F, E, e_2, p_1, ρ
Initialize : x_1^0, ξ_2^0, z_1^0, u_1^0
k = 0
while convergence criterion is not satisfied do
    x_1^{k+1} = argmin_{x_1} ( (1/2)(x_1^T F^T F x_1) + (y^k)^T (E x_1)
    + (ρ/2) ||E x_1 − ξ_2^k + z_1^k + e_2||_2^2 )
    ξ_2^{k+1} = ξ_{2 p_1/ρ}(v), where v = E x_1^{k+1} + z_1^k + e_2 + u_1^k
    for i = 1 : m do
        if v(i) > p_1/ρ then
            ξ_2^{k+1}(i) = v(i) − p_1/ρ
        else if |v(i)| ⩽ p_1/ρ then
            ξ_2^{k+1}(i) = 0
        else
            ξ_2^{k+1}(i) = v(i) + p_1/ρ
        end if
    end for
    z_1^{k+1} = ξ_2^{k+1} − (E x_1^{k+1} + e_2 + u_1^k)
    for j = 1 : n do
        if z_1^{k+1}(j) < 0 then
            ξ_2^{k+1}(j) = ξ_2^{k+1}(j) + z_1^{k+1}(j)
            z_1^{k+1}(j) = 0
        end if
    end for
    u_1^{k+1} = u_1^k + E x_1^{k+1} − ξ_2^{k+1} + z_1^{k+1} + e_2
    k = k + 1
end while
```

Similarly to the TWSVM 1st plane, the optimization problem for the TWSVM 2nd plane follows the same formulation and methodology.

Consider (17)

$$x_2^{k+1} = \arg\min_{x_2} \left(L_\rho \left(x_2, \xi_1^k, z_2^k, y_2^k \right) \right),$$

$$= \arg\min_{x_2} \left(\frac{1}{2}\left(x_2^T E^T E x_2 \right) + \left(y_2^k \right)^T \left(F x_2 \right) + \frac{\rho}{2} \left\| F x_2 + \xi_1^k - z_2^k - e_1 \right\|_2^2 \right).$$

Following the approach used for the 1st plane, we employ the Interior Point Method (IPM) to efficiently solve the argmin problem for the 2nd plane, taking advantage of its robustness in handling constrained optimization.

Next, ξ_1^k is solved by Proximal operators.

Consider (18)

$$\xi_1^{k+1} = \arg\min_{\xi_1} \left(L_\rho \left(x_2^{k+1}, \xi_1, z_2^k, y_2^k \right) \right), \ \xi_1 \geq 0,$$

$$= \arg\min_{\xi_1} \left(\left\| \xi_1 \right\|_1 + \frac{1}{2\frac{p_2}{\rho}} \left\| \zeta_1 - \left(-F x_2^{k+1} + z_2^k + e_1 - \frac{y_2^k}{\rho} \right) \right\|_2^2 \right), \ \xi_1 \geq 0.$$

By using Proximal operator.

Then, we get the form of ξ_1^{k+1} as follow:

$$\xi_1^{k+1} = \xi_{1\frac{p_2}{\rho}}(w) = \begin{cases} w - \frac{p_2}{\rho} & , w > \frac{p_2}{\rho}, \\ 0 & , |w| \leq \frac{p_2}{\rho}, \\ w + \frac{p_2}{\rho} & , w < -\frac{p_2}{\rho}, \end{cases}$$

where $w = -F x_2^{k+1} + z_2^k + e_1 - \frac{y_2^k}{\rho}$.

Next, z_2^{k+1} is solved by differentiating with respect to its parameters.

Consider (19)

$$z_2^{k+1} = \arg\min_{z_2} \left(L_\rho \left(x_2^{k+1}, \xi_1^{k+1}, z_2, y_2^k \right) \right), \ z_2 \geq 0,$$

$$= \xi_1^{k+1} + F x_2^{k+1} + \frac{y_2^k}{\rho} - e_1, \ z_2 \geq 0.$$

Similarly, for z_2, the projection $z = \max(0, z)$ is applied to satisfy $z_2 \geq 0$. Any removed component of z_2 is compensated by updating ξ_1 to maintain constraint satisfaction.

In the last step, y_2^{k+1} is update as

$$y_2^{k+1} = y_2^k + \rho\left(F x_2^{k+1} + \xi_1^{k+1} - z_2^{k+1} - e_1 \right).$$

To simplify the process, we will let $\frac{y_2^k}{\rho} = u_2^k$.

Then, we get $u_2^{k+1} = u_2^k + F x_2^{k+1} + \xi_1^{k+1} - z_2^{k+1} - e_1$.

Thus, ADMM proceed iterations for (12) as follow:

$$x_2^{k+1} = \arg\min_{x_2} \left(\frac{1}{2}\left(x_2^T E^T E x_2 \right) + \left(y_2^k \right)^T \left(F x_2 \right) + \frac{\rho}{2} \left\| F x_2 + \xi_1^k - z_2^k - e_1 \right\|_2^2 \right),$$

$\xi_1^{k+1} = \xi_{1\frac{p_2}{\rho}}(w)$, where $w = -Fx_2^{k+1} + z_2^k + e_1 - u_2^k$,

$z_2^{k+1} = \xi_1^{k+1} + Fx_2^{k+1} + u_2^k - e_1,\ z_2 \geq 0$,

$u_2^{k+1} = u_2^k + Fx_2^{k+1} + \xi_1^{k+1} - z_2^{k+1} - e_1.$

The pseudocode of the model for TWSVM 2nd plane is provide in Algorithm 2.

Algorithm 2 ADMM with IPM on TWSVM 2nd plane

Input : F, E, e_1, p_2, ρ
Initialize : $x_2^0, \xi_1^0, z_2^0, u_2^0$
$k = 0$
while convergence criterion is not satisfied do
  $x_2^{k+1} = \text{argmin}_{x_2}(\frac{1}{2}(x_2^T E^T E x_2) + (y_2^k)^T(Fx_2) + \frac{\rho}{2}\left\|Fx_2 + \xi_1^k - z_2^k - e_1\right\|_2^2)$
  $\xi_1^{k+1} = \xi_{1\frac{p_2}{\rho}}(w)$, where $w = -Fx_2^{k+1} + z_2^k + e_1 - u_2^k$
  for $i = 1 : m$ do
    if $w(i) > \frac{p_2}{\rho}$ then
      $\xi_1^{k+1}(i) = w(i) - \frac{p_2}{\rho}$
    else if $|w(i)| \leqslant \frac{p_2}{\rho}$ then
      $\xi_1^{k+1}(i) = 0$
    else
      $\xi_1^{k+1}(i) = w(i) + \frac{p_2}{\rho}$
    end if
  end for
  $z_2^{k+1} = \xi_1^{k+1} + (Fx_2^{k+1} - e_1 + u_2^k)$
  for $j = 1 : n$ do
    if $z_2^{k+1}(j) < 0$ then
      $\xi_1^{k+1}(j) = \xi_1^{k+1}(j) + z_2^{k+1}(j)$
      $z_2^{k+1}(j) = 0$
    end if
  end for
  $u_2^{k+1} = u_2^k + Fx_2^{k+1} + \xi_1^{k+1} - z_2^{k+1} + e_1$
  $k = k + 1$
end while

3.2 ADMM on TWSVM

In This section, we describe how x^{k+1} can be formulated in closed form, with another subproblem solved similarly to the previous sections.

Consider (13)

$$x_1^{k+1} = \arg\min_{x_1}\left(L_\rho\left(x_1, \xi_2^k, z_1^k, y_1^k\right)\right),$$

$$= \arg\min_{x_1}\left(\frac{1}{2}\left(x_1^T F^T F x_1\right) + \left(y_1^k\right)^T\left(Ex_1\right) + \frac{\rho}{2}\left\|Ex_1 - \xi_2^k + z_1^k + e_2\right\|_2^2\right).$$

By differentiating x_1^{k+1} with respect to x_1.

Then, we get

$$\frac{\partial L_\rho\left(x_1,\xi_2^k,z_1^k,y_1^k\right)}{\partial x_1}=0,\quad \Rightarrow\ F^T F x_1^{k+1}+E^T y_1^k+\rho E^T\left(Ex_1^{k+1}-\xi_2^k+z_1^k+e_2\right)=0,$$

$$F^T F x_1^{k+1}+E^T y_1^k+\rho E^T E x_1^{k+1}-\rho E^T\left(\xi_2^k-z_1^k-e_2\right)=0,$$

$$\left(F^T F+\rho E^T E\right)x_1^{k+1}=\rho E^T\left(\xi_2^k-z_1^k-e_2\right)-E^T y_1^k,$$

$$x_1^{k+1}=\left(F^T F+\rho E^T E\right)^{-1}\left[\rho E^T\left(\xi_2^k-z_1^k-e_2-\frac{y_1^k}{\rho}\right)\right].$$

Next, ξ_2^{k+1} is solved by Proximal operators.

Consider (14)

$$\xi_2^{k+1}=\arg\min_{\xi_2}\left(L_\rho\left(x_1^{k+1},\xi_2,z_1^k,y_1^k\right)\right),\ \xi_2\geq 0\ ,$$

$$=\arg\min_{\xi_2}\left(\left\|\xi_2\right\|_1+\frac{1}{2\frac{p_1}{\rho}}\left\|\xi_2-\left(Ex_1^{k+1}+z_1^k\quad+e_2-\frac{y_1^k}{\rho}\right\|_2^2\right),\ \xi_2\geq 0.$$

By using Proximal operator.

Then, we get the form of ξ_2^{k+1} as follow:

$$\xi_2^{k+1}=\xi_{2_{\frac{p_1}{\rho}}}(v)=\begin{cases} v-\dfrac{p_1}{\rho} & ,v>\dfrac{p_1}{\rho},\\ 0 & ,|v|\leq\dfrac{p_1}{\rho},\\ v+\dfrac{p_1}{\rho} & ,v<-\dfrac{p_1}{\rho},\end{cases}$$

where $v=Ex_1^{k+1}+z_1^k+e_2+\dfrac{y_1^k}{\rho}$.

Next, z_1^{k+1} is solved by differentiating with respect to its parameters.

Consider (15)

$$z_1^{k+1}=\arg\min_{z_1}\left(L_\rho\left(x_1^{k+1},\xi_2^{k+1},z_1,y_1^k\right)\right),\ \ z_1\geq 0,$$

$$=\arg\min_{z_1}\left(y_1^k z_1+\frac{\rho}{2}\left\|Ex_1^{k+1}-\xi_2^{k+1}+z_1^k+e_2\right\|_2^2\right),z_1\geq 0,$$

$$\frac{\partial L_\rho\left(x_1^{k+1},\xi_2^{k+1},z_1,y_1^k\right)}{\partial z_1}=0,\ \Rightarrow\ y_1^k+\rho\left(Ex_1^{k+1}-\xi_2^{k+1}+z_1^{k+1}+e_2\right)=0,\ \ z_1\geq 0,$$

$$z_1^{k+1}=\xi_2^{k+1}-\left(Ex_1^{k+1}+e_2+\frac{y_1^k}{\rho}\right),\ z_1\geq 0.$$

Similar to the previous section, we employ a projection step for z_1 to prevent conditions where $z_1\leq 0$, ensuring feasibility of the optimization constraints throughout the iterations.

In the last step, y_1^{k+1} is update as:

$$y_1^{k+1}=y_1^k+\rho\left(Ex_1^{k+1}-\xi_2^{k+1}+z_1^{k+1}+e_2\right).$$

To simplify the process, we will let $\frac{y_1^k}{\rho} = u_1^k$.

Then, we get $u_1^{k+1} = u_1^k + Ex_1^{k+1} - \xi_2^{k+1} + z_1^{k+1} + e_2$.

Thus, ADMM proceed iterations for (11) as follow:

$$x_1^{k+1} = \left(F^T F + \rho E^T E\right)^{-1}\left[\rho E^T\left(\xi_2^k - z_1^k - e_2 - u_1^k\right)\right],$$

$$\xi_2^{k+1} = \xi_{2\frac{p_1}{\rho}}(v), \text{ where } v = Ex_1^{k+1} + z_1^k + e_2 + u_1^k,$$

$$z_1^{k+1} = \xi_2^{k+1} - \left(Ex_1^{k+1} + e_2 + u_1^k\right),\ z_1 \geq 0,$$

$$u_1^{k+1} = u_1^k + Ex_1^{k+1} - \xi_2^{k+1} + z_1^{k+1} + e_2.$$

The pseudocode of the model for TWSVM 1st plane is provide in Algorithm 3.

Algorithm 3 ADMM on TWSVM 1st plane

```
Input : F, E, e_2, p_1, ρ
Initialize : x_1^0, ξ_2^0, z_1^0, u_1^0
k = 0
while convergence criterion is not satisfied do
    x_1^{k+1} = (F^T F + ρE^T E)^{-1}[ρE^T(ξ_2^k − z_1^k − e_2 − u_1^k)]
    ξ_2^{k+1} = ξ_{2 p_1/ρ}(v), where v = Ex_1^{k+1} + z_1^k + e_2 + u_1^k
    for i = 1 : m do
        if v(i) > p_1/ρ then
            ξ_2^{k+1}(i) = v(i) − p_1/ρ
        else if |v(i)| ⩽ p_1/ρ then
            ξ_2^{k+1}(i) = 0
        else
            ξ_2^{k+1}(i) = v(i) + p_1/ρ
        end if
    end for
    z_1^{k+1} = ξ_2^{k+1} − (Ex_1^{k+1} + e_2 + u_1^k)
    for j = 1 : n do
        if z_1^{k+1}(j) < 0 then
            ξ_2^{k+1}(j) = ξ_2^{k+1}(j) + z_1^{k+1}(j)
            z_1^{k+1}(j) = 0
        end if
    end for
    u_1^{k+1} = u_1^k + Ex_1^{k+1} − ξ_2^{k+1} + z_1^{k+1} + e_2
    k = k + 1
end while
```

Similarly to the TWSVM 1st plane, the optimization problem for the TWSVM 2nd plane follows the same formulation and methodology.

Consider (17)

$$x_2^{k+1} = \arg\min_{x_2}\left(L_\rho\left(x_2, \xi_1^k, z_2^k, y_2^k\right)\right),$$

$$= \arg\min_{x_2}\left(\frac{1}{2}\left(x_2^T E^T E x_2\right) + \left(y_2^k\right)^T\left(Fx_2\right) + \frac{\rho}{2}\left\|Fx_2 + \xi_1^k - z_2^k - e_1\right\|_2^2\right).$$

By differentiating x_2^{k+1} with respect to x_2.

Then, we get

$$\frac{\partial L_{\rho}\left(x_2,\xi_1^k,z_2^k,y_2^k\right)}{\partial x_2}=0, \quad \Rightarrow \quad E^T E x_2^{k+1}+F^T y_2^k+\rho F^T\left(F x_2^{k+1}+\xi_1^k-z_2^k-e_1\right)=0,$$

$$E^T E x_2^{k+1}+F^T y_2^k+\rho F^T F x_2^{k+1}-\rho F^T\left(z_2^k+e_1-\xi_1^k\right)=0,$$

$$\left(E^T E+\rho F^T F\right)x_2^{k+1}=\rho F^T\left(z_2^k+e_1-\xi_1^k\right)-F^T y_2^k,$$

$$x_2^{k+1}=\left(E^T E+\rho F^T F\right)^{-1}\left[\rho F^T\left(z_2^k+e_1-\xi_1^k-\frac{y_1^k}{\rho}\right)\right].$$

Next, ξ_1^k is solved by Proximal operators.

Consider (18)

$$\xi_1^{k+1}=\arg\min_{\xi_1}\left(L_{\rho}\left(x_2^{k+1},\xi_1,z_2^k,y_2^k\right)\right),\ \xi_1\geq 0,$$

$$=\arg\min_{\xi_1}\left(\left\|\xi_2\right\|_1+\frac{1}{2\frac{p_2}{\rho}}\left\|\xi_1-\left(-F x_2^{k+1}+z_2^k+e_1-\frac{y_2^k}{\rho}\right\|_2^2\right),\ \xi_1\geq 0.$$

By using Proximal operator.

Then, we get the form of ξ_1^{k+1} as follow

$$\xi_1^{k+1}=\xi_{1\frac{p_2}{\rho}}(w)=\begin{cases} w-\dfrac{p_2}{\rho} & ,w>\dfrac{p_2}{\rho}, \\ 0 & ,\left|w\right|\leq\dfrac{p_2}{\rho}, \\ w+\dfrac{p_2}{\rho} & ,w<-\dfrac{p_2}{\rho}, \end{cases}$$

where $w=-F x_2^{k+1}+z_2^k+e_1-\dfrac{y_2^k}{\rho}$.

Next, z_2^{k+1} is solved by differentiating with respect to its parameters.

Consider (19)

$$z_2^{k+1}=\arg\min_{z_2}\left(L_{\rho}\left(x_2^{k+1},\xi_1^{k+1},z_2,y_2^k\right)\right),\ z_2\geq 0,$$

$$=\xi_1^{k+1}+F x_2^{k+1}+\frac{y_2^k}{\rho}-e_1,\ z_2\geq 0.$$

As in the previous process, a projection is used to ensure $z_2\geq 0$ and to handle cases where this condition is violated. This step keeps the optimization valid and ensures the method works smoothly.

In the last step, y_2^{k+1} is update as

$$y_2^{k+1}=y_2^k+\rho\left(F x_2^{k+1}+\xi_1^{k+1}-z_2^{k+1}-e_1\right).$$

To simplify the process, we will let $\dfrac{y_2^k}{\rho}=u_2^k$.

Then, we get $u_2^{k+1}=u_2^k+F x_2^{k+1}+\xi_1^{k+1}-z_2^{k+1}-e_1$.

Thus, ADMM proceed iterations for (12) as follow:

$$x_2^{k+1}=\left(E^TE+\rho F^TF\right)^{-1}\left[\rho F^T\left(z_2^k+e_1-\xi_1^k-\frac{y_1^k}{\rho}\right)\right].$$

$$\xi_1^{k+1}=\xi_{1\frac{p_2}{\rho}}(w)\text{, where } w=-Fx_2^{k+1}+z_2^k+e_1-u_2^k,$$

$$z_2^{k+1}=\xi_1^{k+1}+Fx_2^{k+1}+u_2^k-e_1,\ z_2\geq 0,$$

$$u_2^{k+1}=u_2^k+Fx_2^{k+1}+\xi_1^{k+1}-z_2^{k+1}-e_1.$$

The pseudocode of the model for TWSVM 2nd plane is provide in Algorithm 4.

Algorithm 4 ADMM on TWSVM 2nd plane

```
Input : F, E, e_1, p_2, ρ
Initialize : x_2^0, ξ_1^0, z_2^0, u_2^0
k = 0
while convergence criterion is not satisfied do
    x_2^{k+1} = (E^T E + ρF^T F)^{-1}[ρE^T(ξ_1^k − z_2^k − e_1 − u_2^k)]
    ξ_1^{k+1} = ξ_{1 p_2/ρ}(w), where w = −F x_2^{k+1} + z_2^k + e_1 − u_2^k
    for i = 1 : m do
        if w(i) > p_2/ρ then
            ξ_1^{k+1}(i) = w(i) − p_2/ρ
        else if |w(i)| ⩽ p_2/ρ then
            ξ_1^{k+1}(i) = 0
        else
            ξ_1^{k+1}(i) = w(i) + p_2/ρ
        end if
    end for
    z_2^{k+1} = ξ_1^{k+1} + (F x_2^{k+1} − e_1 + u_2^k)
    for j = 1 : n do
        if z_2^{k+1}(j) < 0 then
            ξ_1^{k+1}(j) = ξ_1^{k+1}(j) + z_2^{k+1}(j)
            z_2^{k+1}(j) = 0
        end if
    end for
    u_2^{k+1} = u_2^k + F x_2^{k+1} + ξ_1^{k+1} − z_2^{k+1} + e_1
    k = k + 1
end while
```

3.3 Robust Twin Support Vector Machine (*RTWSVM*)

In this section, we introduce an adjustment to the Twin Support Vector Machine (TWSVM) by modifying the parameter e which traditionally controls the tolerance for misclassifications in the model. let reconsider the constraints of TWSVM optimization problem:

$$-\left(Bw_1+e_2b_1\right)+\xi_2\geq e_2,\ \xi_2\geq 0,$$

$$\left(Aw_2+e_1b_2\right)+\xi_1\geq e_1,\ \xi_1\geq 0.$$

In standard TWSVM, e is used to allow some flexibility in separating the classes while balancing the margin and classification error.

To enhance the model's ability to better separate the classes, we propose the fuzzy inequalities. Thus, the constraints of TWSVM are rewritten as:

$$-\left(Bw_1+e_2b_1\right)+\xi_2\gtrsim e_2,\ \xi_2\geq 0,$$

$$\left(Aw_2+e_1b_2\right)+\xi_1\gtrsim e_1,\ \xi_1\geq 0,$$

where the symbol $\gtrsim$ indicates that some degree of constraint violation is permitted.

Then, The TWSVM can be formulated as follows:

$$\min_{w_1,b_1,\xi_2} \ \frac{1}{2}\|Aw_1+e_1b_1\|_2^2+\frac{c_1}{2}\|\xi_2\|_1, \tag{21}$$
$$s.t. \ \ -(Bw_1+e_2b_1)+\xi_2 \gtrsim e_2,\ \ \xi_2 \geq 0,$$

$$\min_{w_2,b_2,\xi_1} \ \frac{1}{2}\|Bw_2+e_2b_2\|_2^2+\frac{c_2}{2}\|\xi_1\|_1, \tag{22}$$
$$s.t. \ \ (Aw_2+e_1b_2)+\xi_1 \gtrsim e_1,\ \ \xi_1 \geq 0.$$

To address the optimization, we employ the method introduced by [Sabzekar, M., & Hasheminejad, S. M. H.][7], which has proven effective in similar problems.

Thus, the optimization problem (21) and (22) are converted to:

$$\min_{w_1,b_1,\xi_2} \ \frac{1}{2}\|Aw_1+e_1b_1\|_2^2+\frac{c_1}{2}\|\xi_2\|_1, \tag{23}$$
$$s.t. \ \ -(Bw_1+e_2b_1)+\xi_2 \geq e_2+d(1-\alpha),\ \ \xi_2 \geq 0,$$

$$\min_{w_2,b_2,\xi_1} \ \frac{1}{2}\|Bw_2+e_2b_2\|_2^2+\frac{c_2}{2}\|\xi_1\|_1, \tag{24}$$
$$s.t. \ \ (Aw_2+e_1b_2)+\xi_1 \geq e_1+d(1-\alpha),\ \ \xi_1 \geq 0,$$

where $d(1-\alpha)$ is a new positive parameter that provides additional flexibility in the margin expansion. This modification allows for more control over the trade-off between the margin size and the acceptable level of misclassification.

The adjustment of parameters d and α plays a critical role in balancing margin expansion and misclassification tolerance. Larger values of α result in more aggressive margin expansion, which is particularly beneficial for datasets with class imbalance, as minority class samples are better separated.

Conversely, d introduces flexibility in handling noisy datasets by relaxing the constraints.

IV. Numerical result

To evaluate the performance of the proposed methods, we conducted experiments on various datasets to compare the F-measure and accuracy of our algorithms. All experiments were conducted using 5-fold cross-validation. The datasets utilized in the experiments span different domains and dimensions, ensuring comprehensive evaluation. The best performance in each dataset is highlighted in bold.

Dataset		ADMM-TWSVM	ADMM-IPM-TWSVM	LSTWSVM	TWSVM
Breast Cancer Wisconsin (558 × 5)	F-measure	92.44	**93.00**	90.85	91.53
	Accuracy	94.77	**95.13**	94.05	93.42
Dry Bean (5573 × 5)	F-measure	94.70	**95.96**	94.13	93.87
	Accuracy	95.92	**96.94**	95.96	94.79
Heart Disease (303 × 13)	F-measure	84.68	83.08	**86.49**	85.42
	Accuracy	82.66	81.17	**84.33**	83.23
Electricity (36250 × 7)	F-measure	46.10	40.7	**59.41**	60.54
	Accuracy	68.07	67.03	**72.56**	71.54
Pima indians diabetes (768 × 8)	F-measure	63.42	**64.57**	59.88	61.45
	Accuracy	72.97	72.68	**74.90**	72.45
Mozilla 4 (12436 × 5)	F-measure	85.88	**86.18**	86.05	85.51
	Accuracy	**80.79**	80.60	80.37	79.75
Satellite (5100 × 38)	F-measure	**63.83**	63.07	54.71	55.25
	Accuracy	98.64	98.58	**99.03**	98.50
Magic Telescope (15216 × 11)	F-measure	84.20	83.51	**84.58**	83.59
	Accuracy	77.29	76.14	**78.57**	77.04

Fig. 1. Comparative performance of ADMM-TWSVM, ADMM-IPM-TWSVM, LSWTSVM and TWSVM on various datasets. The table reports the F-measure and accuracy for each algorithm.

4.1 Experiment on ADMM-TWSVM and ADMM-IPM-TWSVM

On Figure 1 presents a comparison of F-measure and Accuracy between the proposed algorithm, LSTSVM, and TWSVM. The results demonstrate the improved classification performance of the proposed method, achieving higher accuracy and a better balance between precision and recall, as reflected in the F-measure, compared to the traditional algorithms.

- The ADMM-IPM-TWSVM consistently outperforms in most datasets, achieving the highest F-measure and accuracy in datasets such as Breast Cancer Wisconsin, Dry Bean, and Mozilla 4.
- The hybrid approach combining ADMM and IPM leverages the efficiency of ADMM for optimization and the precision of IPM, enabling robust performance across a range of dataset sizes and complexities.
- LSTWSVM often demonstrates strong performance, particularly in high-dimensional datasets like Satellite and Electricity, where it achieves the best accuracy in some cases.

TWSVM, while computationally efficient, tends to underperform compared to the proposed methods in datasets with noise or complex feature distributions.

Dataset		ADMM-RTWSVM	ADMM-IPM-RTWSVM	LSTWSVM	TWSVM
Breast Cancer Wisconsin (558×5)	F-measure	92.44	**93.00**	90.85	91.53
	Accuracy	94.77	**95.13**	94.05	93.42
Dry Bean (5573×5)	F-measure	94.70	**95.96**	94.13	93.87
	Accuracy	95.92	**96.94**	95.96	94.79
Heart Disease (303×13)	F-measure	84.68	83.08	**86.49**	85.42
	Accuracy	82.66	81.17	**84.33**	83.23
Electricity (36250×7)	F-measure	**62.93**	61.49	59.41	60.54
	Accuracy	68.51	70.06	**72.56**	71.54
Pima indians diabetes (768×8)	F-measure	65.10	**65.32**	59.88	61.45
	Accuracy	73.33	72.81	**74.90**	72.45
Mozilla 4 (12436×5)	F-measure	87.15	**87.75**	86.05	85.51
	Accuracy	81.64	**83.56**	80.37	79.75
Satellite (5100×38)	F-measure	71.58	**72.44**	54.71	55.25
	Accuracy	99.33	**99.35**	99.03	98.50
Magic Telescope (15216×11)	F-measure	**84.64**	83.94	84.58	83.59
	Accuracy	**78.92**	77.56	78.57	77.04

Fig. 2. Comparative performance of ADMM-RTWSVM, ADMM-IPM-RTWSVM, LSWTSVM and TWSVM on various datasets. The table reports the F-measure and accuracy for each algorithm.

4.2 Experiment on ADMM-RTWSVM and ADMM-IPM-RTWSVM

On Figure 2 compares the performance of the proposed methods, ADMM-RTWSVM and ADMM-IPM-RTWSVM, with baseline algorithms, LSTWSVM and TWSVM, across several datasets. Additionally, the impact of adjusting the parameters d and α in the RTWSVM formulation is discussed.

The adjustments to the parameters d and α in RTWSVM have resulted in improvements in the performance of both proposed algorithms (ADMM-RTWSVM and ADMM-IPM-RTWSVM) across most datasets. By fine tuning these parameters, we enhance the balance between margin expansion and misclassification tolerance, leading to better classification results, particularly in datasets with noise or class imbalances.

In datasets with higher noise or class imbalance, such as Pima Indians Diabetes and Mozilla 4 and Satellite, the adjustments to d and α have a clear positive effect. In these cases, ADMM-IPM-RTWSVM achieves higher F-measure and accuracy compared to the baseline methods, showing its ability to better handle more complex data.

However, for datasets like Breast Cancer Wisconsin, Dry Bean, and Heart Disease, where noise is minimal or the dataset is well-behaved, adjusting d and α does not significantly affect the results. In these cases, the original margin and misclassification tolerance settings already provide satisfactory classification performance. Therefore, the impact of adjusting d and α is less pronounced, as the data does not require additional flexibility in handling noise or imbalance.

V. CONCLUSION

This research presents a novel hybrid optimization framework that integrates the Alternating Direction Method of Multipliers (ADMM) and the Interior Point Method (IPM) to address the limitations of Twin Support Vector Machines (TWSVM). By combining the computational efficiency of ADMM with the optimization precision of IPM, the proposed approach achieves significant improvements in handling large-scale and noisy datasets. Additionally, the development of a robust variant, the Robust Twin Support Vector Machine (RTWSVM), further enhances classification performance in challenging scenarios, particularly those involving noise and class imbalances.

Experimental results across various datasets demonstrate the effectiveness of the hybrid ADMM-IPM approach and the robustness introduced by the RTWSVM formulation. Key improvements include higher accuracy and F-measure scores, as well as better generalization capabilities compared to traditional TWSVM and other baseline methods. Furthermore, the introduction of adjustable parameters d and α provides flexibility in balancing margin expansion and misclassification tolerance, tailoring the model to specific dataset characteristics.

The findings of this study underscore the potential of hybrid optimization techniques in advancing classification algorithms, particularly in real-world applications. Future work will focus on further refining the robust formulation and exploring its applications in other machine learning tasks.

REFERENCES

[1] KBoyd, S., & Vandenberghe, L. (2004). Convex optimization. Cambridge university press.J. U. Duncombe, “Infrared navigation—Part I: An assessment of feasibility,” IEEE Trans. Electron Devices, vol. ED-11, pp. 34-39, Jan. 1959.
[2] Wright, S. J. (2006). Numerical optimization.
[3] Wright, S. J. (1997). Primal-dual interior-point methods. Society for Industrial and Applied Mathematics.
[4] Boyd, S., Parikh, N., Chu, E., Peleato, B., & Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine learning, 3(1), 1-122.
[5] Cortes, C. (1995). Support-Vector Networks. Machine Learning.
[6] Khemchandani, R., & Chandra, S. (2007). Twin support vector machines for pattern classification. IEEE Transactions on pattern analysis and machine intelligence, 29(5), 905-910.
[7] Sabzekar, M., & Hasheminejad, S. M. H. (2021). Robust regression using support vector regressions. Chaos, Solitons & Fractals, 144, 110738.
[8] Chang, C. C. (2001). A library for support vector machines. http://www. csie. ntu. edu. tw/~ cjlin/libsvm.
[9] Grandvalet, Y., & Canu, S. (2002). Adaptive scaling for feature selection in SVMs. Advances in neural information processing systems, 15.
[10] Ruppert, D. (2004). The elements of statistical learning: data mining, inference, and prediction.
[11] Kowalski, M., & Torrésani, B. (2009). Sparsity and persistence: mixed norms provide simple signal models with dependent coefficients. Signal, image and video processing, 3, 251-264.
[12] Parikh, N., & Boyd, S. (2014). Proximal algorithms. Foundations and trends® in Optimization, 1(3), 127-239.
[13] Suykens, J. A., & Vandewalle, J. (1999). Least squares support vector machine classifiers. Neural processing letters, 9, 293-300.
[14] Xu, H., & Mannor, S. (2012). Robustness and generalization. Machine learning, 86, 391-423.
[15] UCI Machine Learning Repository. UCI Machine Learning Repository. Retrieved from https://archive.ics.uci.edu/ml/index.php
[16] Kaggle. Kaggle Datasets. Retrieved from Find Open Datasets and Machine Learning Projects | Kaggle
[17] MathWorks. (2024). MATLAB version R2024a. Natick, Massachusetts: The MathWorks, Inc. Retrieved from https://www.mathworks.com