B. TECH. PROJECT REPORT

ON

Sparse Pinball Twin Support Vector Machine and its Large Scale Variant

By:

RAHUL CHOUDHARY and SANCHIT JALAN

DISCIPLINE OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

December, 2018

Sparse Pinball Twin Support Vector Machine and its Large Scale Variant

A PROJECT REPORT

Submitted in partial fulfillment of the requirements
for the award of the degree of Bachelor of Technology

in the

Discipline of Computer Science and Engineering

Submitted by:
RAHUL CHOUDHARY (150001027) and SANCHIT JALAN (150002029)

Guided by:
Dr. ARUNA TIWARI
Associate Professor, Discipline of Computer Science and Engineering, IIT Indore
Dr. M. TANVEER
Assistant Professor, Discipline of Mathematics, IIT Indore

INDIAN INSTITUTE OF TECHNOLOGY INDORE

December, 2018

Declaration of Authorship

We hereby declare that the project entitled **"Sparse Pinball Twin Support Vector Machine and its Large Scale Variant"** submitted in partial fulfillment for the award of the degree of Bachelor of Technology in the Discipline of Computer Science and Engineering and completed under the supervision of **Dr. ARUNA TIWARI**, Associate Professor, Discipline of Computer Science and Engineering, IIT Indore and **Dr. M. TANVEER**, Assistant Professor, Discipline of Mathematics, IIT Indore is an authentic work.

Furthermore, we declare that we have not submitted this work for the award of any other degree elsewhere.

Signature:

_____

Date:

_____

Certificate

This is to certify that the thesis entitled "*Sparse Pinball Twin Support Vector Machine and its Large Scale Variant*" and submitted by Rahul Choudhary, Roll No. 150001027 and Sanchit Jalan, Roll No. 150002029, in partial fulfillment of the requirements for CS 493 B.Tech Project embodies the work done by them under our supervision. It is certified that the declaration made by the students is correct to the best of our knowledge.

Supervisor

Dr. ARUNA TIWARI
Associate Professor,
Indian Institute of Technology Indore
Date:

Supervisor

Dr. M. TANVEER
Assistant Professor,
Indian Institute of Technology Indore
Date:

"Education is not preparation for life; education is life itself. "

John Dewey

Abstract

Sparse Pinball Twin Support Vector Machine and its Large Scale Variant

The original twin support vector machine (TWSVM) formulation works by solving two smaller quadratic programming problems as compared to the traditional hinge-loss SVM (C-SVM) which solves a single large quadratic programming problem - this makes the TWSVM training and testing process faster than the C-SVM. However, these TWSVM problems are based on the hinge-loss function and, hence, are sensitive to feature noise and unstable for resampling. The pinball-loss function, on the other hand, works by minimizing quantile distances which grants noise insensitivity but this comes at the cost of losing sparsity by penalizing correctly classified points as well. To overcome the limitations of TWSVM, we propose a novel sparse pinball twin support vector machine (SPTWSVM) based on the ϵ-insensitive zone pinball loss function to rid the original TWSVM of its noise insensitivity and ensure that the resulting TWSVM problems retain sparsity which makes computations relating to predictions just as fast as the original TWSVM. We further investigate the properties of our model including sparsity, noise insensitivity, and scatter minimization. Exhaustive testing on several benchmark datasets demonstrates that our SPTWSVM is noise insensitive, retains sparsity and, in most cases, outperforms the results obtained by the original TWSVM.

In a quest for further improvement, we extend our first work by making our model feasible for large scale datasets. This is achieved by tweaking our SPTWSVM model, that is adding an extra equality constraint in the primal problem, to avoid calculating large inverse matrices in the Wolfe dual problems. Also, in the same model we add a regularization term to the objective function of SPTWSVM which incorporates the structural risk minimization principle in the second work. Henceforth, the resulting second model, named improved sparse pinball twin support vector machine (ISPTWSVM), retains all attractive properties of the first model while being a viable option for huge real world datasets.

Acknowledgements

We would like to thank our B.Tech Project supervisors **Dr. Aruna Tiwari** and **Dr. M. Tanveer** for their guidance and constant support in structuring the project and their valuable feedback throughout the course of this project. Their overseeing the project meant there was a lot that we learnt while working on it. We thank them for their time and efforts.

Most importantly, we are thankful for each other's camaraderie, without which writing the thesis would have taken much longer. Also, we are thankful to our other friends who were a constant source of both motivation and light hearted humour.

We are really grateful to the Institute for the opportunity to be exposed to systemic research, especially, Dr. Aruna Tiwari's lab for providing the necessary hardware utilities to complete the project.

Lastly, we offer our sincere thanks to everyone who helped us to complete this project, whose name we might have forgotten to mention.

Contents

List of Tables

List of Figures

List of Abbreviations

SV	**S**upport **V**ector
SVM	**S**upport **V**ector **M**achine
TWSVM	**T**win **S**upport **V**ector **M**achines
TBSVM	**T**win **B**ounded **S**upport **V**ector **M**achines
SPTWSVM	**S**parse **P**inball **T**win **S**upport **V**ector **M**achine
ISPTWSVM	**I**mproved **S**parse **P**inball **T**win **S**upport **V**ector **M**achine
UCI	**U**niversity of **C**alifornia **I**rvine
SMO	**S**equential **M**inimal **O**ptimization
SOR	**S**uccessive **O**ver **R**elaxation
Quadprog	**Q**uadratic **P**rogramming
QPP	**Q**uadratic **P**rogramming **P**roblem

Dedicated to my wonderful parents, for instilling into me the love for ventures into the scientific unknown, and for placing unwavering belief in me. - Rahul Choudhary

Dedicated to my beloved parents for being a constant source of inspiration and guidance to me. - Sanchit Jalan

Chapter 1

Introduction

1.1 Background

Support Vector Machines (SVMs), a supervised machine learning model was originally proposed by Vapnik [1], [2] for binary classification. Since then SVMs have been extended to solve multi-classification problems [3], [4], [5], [6], [7] as well. The effectiveness of SVMs have led them to be widely applied to a large spectrum of research areas such as face recognition [8], bio medicine [9], text recognition [10], brain computer interface [11], [12] and cancer recognition [13]. The basic idea of SVM problem is to find an optimal separating hyperplane between two classes which maximizes the distance from the convex hull of each class. Furthermore, SVMs implement the structural risk minimization (SRM) principle that minimizes the upper bound of generalization error. Solving the SVM primal problem involves minimization of a convex quadratic function subject to linear inequality constraints where the main challenge is the high computational complexity of training, i.e. $O(m^3)$, where m is the total number of the training samples.

In contrast to the aforementioned idea of generating two parallel supporting hyperplanes in SVMs, Mangasarian and Wild [14] introduced a generalized eigen-value proximal SVM (GEPSVM) which generates two non-parallel hyperplanes for binary classification problems. Subsequently, Jayadeva et al. [15] proposed a twin support vector machine (TWSVM) which generates two non-parallel hyperplanes by solving two smaller-sized QPPs such that each hyperplane is as close as possible to one class and as far as possible from the other. The main idea behind solving two small QPPs rather than a single large QPP is to speed up the learning approximately by four times as compared to the classical SVM (C-SVM). Henceforth, TWSVM is very effective in dealing with datasets containing a large number of samples, where it is simply ineffective to apply the standard SVM.

Despite the speedup it offers, the TWSVM is based on hinge loss and is sensitive to noise and unstable for re-sampling. To overcome this limitation of the hinge loss, Huang et al. [16] proposed an interesting approach in which they introduce the pinball loss ($L_\tau(u)$) to C-SVM for the first time. Pinball loss is based on the idea of maximizing the quantile distance between two classes instead of maximizing the distance between the closest samples of the two classes. This property, in essence, introduces noise insensitivity as well as resampling stability into the C-SVM. However, introducing pinball loss to the C-SVM leads to the solution losing its sparsity. In order to maintain the sparsity in the Pin-SVM, Huang et al. [16] introduced an ϵ-insensitive zone into the pinball loss ($L^{\epsilon}_{\tau}(u)$). This sparse pinball loss model is noise insensitive as well as sparse in the solution obtained. However, Sparse Pin SVM, as compared to the TWSVM, still needs to solve a single large QPP, i.e. it entails a higher time complexity. Recently, several modifications have been done to the TWSVM in an attempt to enhance its time complexity and performance [17], [18], [19], [20], [21, 22, 23, 24, 25], [26], [27] and extend the TWSVM from binary class to multi-class classification [28], [29], [30] but none of these approaches address the issues of insensitivity to noise around the decision boundary and ensuring sparsity of the

solution. Hence, there is a need to introduce noise insensitivity and sparsity into the TWSVM formulation.

Inspired and motivated by the studies of twin support vector machine (TWSVM) [15] and Sparse Pinball SVM [16], we propose a novel Sparse Pinball Twin Support Vector Machine (SPTWSVM) which has numerous advantages. First, SPTWSVM is insensitive to noise while retaining the sparsity of solution as compared to TWSVM. Second, our SPTWSVM is faster than the Pin-SVM since it solves two smaller QPPs and, thus, SPTWSVM performs predictions in remarkably less computational time. Third, numerical experiments show that SPTWSVM outperforms the classification accuracy of the TWSVM and Sparse Pin SVM in most cases. Fourth, unlike in the other relevant existing work, [31], Xu et al. introduce the pinball loss into TPMSVM which does not have the generalized TWSVM formulation. On the other hand, the novel SPTWSVM model retains the original TWSVM form and can easily accommodate other models built upon TWSVM.

However, TWSVMs do not minimize the structural risk in its formulation, minimizing the empirical risk instead. Additionally, TWSVMs have to calculate inverse matrices in the dual problem with extra assumptions. These drawbacks were addressed by Shao et al. [32] where they proposed the twin bounded support vector machines (TBSVM). Despite these improvements, TBSVM has to compute inverse matrices in the dual problems, which is in practice intractable for a large dataset. Furthermore, in TWSVMs and TBSVMs, the non-linear case with the linear kernel is not equivalent to the linear case. In other words, TWSVMs and TBSVMs cannot replicate the exact behaviour of the linear case when using a non-linear kernel. As a result, they can only solve an approximate formulation whereas in standard SVMs, one problem is solved for both the cases using different kernels. These limitations of TWSVM and TBSVM render them inferior to the traditional SVM problem and prevent them from being applied in real applications. We attempt to introduce further changes in the primal problems' objective functions of the TBSVM which make it suitable for large scale datasets. We label our improved sparse pinball twin support vector machine for large scale datasets as ISPTWSVM.

The rest of the project report is organized as follows: the relevant background information concerning TWSVM, Sparse Pinball SVM and TBSVM is in Chapter 2. In Chapter 3, we explain the objectives of the project. In Chapter 4, we introduce and explain the formulations of SPTWSVM and ISPTWSVM for both the linear and non-linear cases. In Chapter 5, we discuss some properties and relevant results of our proposed SPTWSVM. In Chapter 6, we present significant experimental results obtained when SPTWSVM and ISPTWSVM are applied on benchmark and synthetic datasets, which clearly demonstrate the feasibility and effectiveness of our two novel models. In Chapter 7, we complete the project report with concluding remarks.

Chapter 2

Literature Survey

The following chapter discusses literature pertaining to previously known methods of incorporating noise-insensitivity into the conventional SVM formulation, the twin support vector machine formulation, and the improved twin support vector machine for large scale datasets. It describes in detail how the pinball loss function can impart noise insensitivity to SVM models and how it can be altered to retain sparsity which is otherwise lost with the original pinball loss. The readers are referred to [16, 15] for more details.

2.1 Sparse Pinball Support Vector Machine

Huang et al. [16] introduced the pinball loss SVM (pin-SVM) formulation which brought noise insensitivity to the SVM classifier. Consider a binary dataset $\mathbf{z} = \{x_i, y_i\}_{i=1}^m$ where $x_i \in \mathbb{R}^n$ and $y_i \in \{-1, 1\}$. Then, the pin-SVM problem is as follows:

$$\min_{w,b} \quad \frac{1}{2}w^T w + C\sum_{i=1}^{m} L_\tau(1 - y_i(w^T x_i + b)).$$

Here, $w \in \mathbb{R}^n$ and $b \in \mathbb{R}$ are the weight vector and bias, respectively, which define the hyperplane $\mathcal{H} : w^T x + b = 0$, C is a constant and L_τ is the pinball loss function. The decision function of the above formulation is based on the sign of $w^T x + b$: x is assigned to class +1 if the value is positive otherwise it is assigned to class -1. This formulation, unlike the hinge loss SVM, enables the classifier not to be influenced by feature noise near the decision boundary. The way this model works is by penalizing correctly classified samples as well, which is evident when we take a look at the pinball loss function:

$$L_\tau(u) = \begin{cases} u, & u \geq 0, \\ -\tau u, & u < 0. \end{cases}$$

The above pinball loss function can be regarded as a generalized ℓ_1 loss. When $u < 0$ we get an error value not equal to zero. Thus, we get weight vector w as a linear combination of vectors that lie not just near the decision boundary but away from the boundary as well. As a result, pin-SVM approximates a model which maximizes the quantile distance between two classes.

However, noise insensitivity gained from pinball loss leads to losing sparsity of solution. This is because the pinball loss function's sub-gradient is non-zero almost everywhere. To remediate this shortcoming, the authors in [16] suggest using an ϵ-insensitive pinball loss function:

$$L_\tau^\epsilon(u) = \begin{cases} u - \epsilon, & u > \epsilon, \\ 0, & -\frac{\epsilon}{\tau} \leq u \leq \epsilon, \\ -\tau(u + \frac{\epsilon}{\tau}), & u < -\frac{\epsilon}{\tau}. \end{cases} \tag{1}$$

The sub-gradient of the above loss function turns out to be zero in the range $[\frac{\epsilon}{\tau}, \epsilon]$ providing sparsity to the model. This way both insensitivity to noise and sparsity can be achieved.

2.2 Twin Support Vector Machine (TWSVM)

Jayadeva et al. [15] proposed the twin support vector machine which formulates two smaller sized quadratic programming problems (QPPs), obtaining two non-parallel hyperplanes corresponding to each of the two classes. Let m_1 and m_2 be the number of samples corresponding to classes +1 and -1, respectively. Further, let $A_{m_1 \times n}$ and $B_{m_2 \times n}$ be the matrices containing the feature vectors of the samples of class +1 and -1, respectively. The aim here is to then derive the following two non-parallel hyperplanes:

$$\mathcal{H}_1 : x^T w^{(1)} + b^{(1)} = 0, \text{ and}$$

$$\mathcal{H}_2 : x^T w^{(2)} + b^{(2)} = 0.$$

Here, $w^{(1)} \in \mathbb{R}^n$ and $b^{(1)} \in \mathbb{R}$ are the weight vector and bias, respectively, of the first hyperplane $\mathcal{H}_1$. Similarly, $w^{(2)} \in \mathbb{R}^n$ and $b^{(2)} \in \mathbb{R}$ are the weight vector and bias, respectively, of the second hyperplane $\mathcal{H}_2$. These planes are arrived at by two QPPs which are similar in formulation to a typical SVM problem.

The QPPs of TWSVM minimize the sum of squares of the distances of the samples of a class (say +1) to its corresponding hyperplane and ensuring that the samples of the other class (say -1) are at least 1 distance away from the hyperplane. A similar QPP is constructed for the other hyperplane. Hence, the two problems are of the form:

$$\begin{aligned} \min_{w^{(1)}, b^{(1)}, \xi} \quad & \frac{1}{2}(Aw^{(1)} + e_1 b^{(1)})^T (Aw^{(1)} + e_1 b^{(1)}) + c_1 e_2^T \xi \\ \text{subject to} \quad & -(Bw^{(1)} + e_2 b^{(1)}) + \xi \geq e_2, \quad \xi \geq 0. \end{aligned} \tag{2a}$$

and

$$\begin{aligned} \min_{w^{(2)}, b^{(2)}, \xi} \quad & \frac{1}{2}(Bw^{(2)} + e_2 b^{(2)})^T (Bw^{(2)} + e_2 b^{(2)}) + c_2 e_1^T \xi \\ \text{subject to} \quad & (Aw^{(2)} + e_1 b^{(2)}) + \xi \geq e_1, \quad \xi \geq 0. \end{aligned} \tag{2b}$$

Here $c_1, c_2 > 0$ are parameters and e_1 and e_2 are vectors of ones of appropriate dimensions. ξ is an error variable that is used to bound the error term (hinge loss in this case). Once the weight vectors and the biases of $\mathcal{H}_1$ and $\mathcal{H}_2$ have been calculated, one can predict the class l, $(l = 1, 2)$ of a new sample $x \in \mathbb{R}^n$ using the following decision function:

$$l = \arg\min_{i=1,2} |x^T w^{(i)} + b^{(i)}|.$$

Here, $|.|$ is the perpendicular distance of the sample x from a given hyperplane.

2.3 Twin Bounded Support Vector Machines (TBSVM)

In an attempt to improve the TWSVM model, Shao et al. [32] proposed the twin bounded support vector machines (TBSVM) where they introduced the structural risk minimization principle in the TWSVM problem and eliminated the need to well condition the matrix (to calculate its inverse) involved in the dual of TWSVM. This is achieved by introducing a regularization term in

the objective function of TWSVM which minimize the structural risk with the idea of maximizing the margin. An added benefit of introducing the regularization term is that it eliminates the need to derive the dual of the problem without any extra assumptions unlike TWSVM. Thus, TBSVM stands as a significant improvement over TWSVM. The TBSVM primal problems are as follows:

$$\begin{aligned} \min_{w^{(1)},b^{(1)},\xi} \quad & \frac{1}{2}c_3(||w^{(1)}||^2 + b^{(1)2}) + \frac{1}{2}(Aw^{(1)} + e_1b^{(1)})^T(Aw^{(1)} + e_1b^{(1)}) + c_1e_2^T\xi \qquad (3a) \\ \text{subject to} \quad & -(Bw^{(1)} + e_2b^{(1)}) + \xi \geq e_2, \quad \xi \geq 0. \end{aligned}$$

and

$$\begin{aligned} \min_{w^{(2)},b^{(2)},\xi} \quad & \frac{1}{2}c_4(||w^{(2)}||^2 + b^{(2)2}) + \frac{1}{2}(Bw^{(2)} + e_2b^{(2)})^T(Bw^{(2)} + e_2b^{(2)}) + c_2e_1^T\xi \qquad (3b) \\ \text{subject to} \quad & (Aw^{(2)} + e_1b^{(2)}) + \xi \geq e_1, \quad \xi \geq 0. \end{aligned}$$

Here, A, B, $w^{(1)}$, $w^{(2)}$, $b^{(1)}$, $b^{(2)}$, e_1, e_2 and ξ are the same as in TWSVM, and c_1, c_2, c_3, c_4 are positive parameters. The introduction of term $\frac{1}{2}c_3(||w^{(1)}||^2 + b^{(1)2})$ in (3a) introduces the structural risk minimization principle since the term corresponds to the distance between the proximal hyperplane, $w^{(1)T}x + b^{(1)} = 0$, and the bounding hyperplane, $w^{(1)T}x + b^{(1)} = -1$. A similar analysis holds for (3b).

For the sake of conciseness, we only consider the dual problem of (3a). Writing its Lagrangian,

$$\begin{aligned} \mathcal{L} = & \frac{1}{2}c_3(||w^{(1)}||^2 + b^{(1)2}) + \frac{1}{2}(Aw^{(1)} + e_1b^{(1)})^T(Aw^{(1)} + e_1b^{(1)}) + c_1e_2^T\xi \\ & -\alpha^T(-(Bw^{(1)} + e_2b^{(1)}) + \xi - e_2) - \beta^T\xi + \mu^T(Aw^{(1)} + e_1b^{(1)} - \eta_1), \end{aligned}$$

where $\alpha \in \mathbb{R}^{m_2}$, $\beta \in \mathbb{R}^{m_2}$, $\mu \in \mathbb{R}^{m_1}$ are Lagrangian multipliers corresponding to the different constraints. After applying the necessary and sufficient K.K.T. conditions, we obtain the Wolfe dual of the first TBSVM problem:

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2}\alpha^TG(H^TH + c_3I)^{-1}G^T\alpha - e_2^T\alpha \\ \text{subject to} \quad & 0 \leq \alpha \leq c_1e_2, \end{aligned}$$

where

$$H = \begin{bmatrix} A & e_1 \end{bmatrix}, \; G = \begin{bmatrix} B & e_2 \end{bmatrix},$$

I is the identity matrix of size $m_1 \times m_1$. As is evident, $(H^TH + c_3I)^{-1}$ is naturally nonsingular and, hence, invertible without making any extra assumptions unlike TWSVM's dual problems. However, despite the differences in formulation, the decision function of TBSVM is similar to that of TWSVM.

Chapter 3

Objectives

There have been few attempts to introduce noise sensitivity near the decision boundary in twin support vector machines. Huang et al. [16] describe an approach where they take the pinball loss, usually applied in regression but not to classification, and use it instead of the hinge loss in the traditional SVM, labelling it Pin-SVM. This method approximates the problem of maximizing the quantile distances between the two classes; quantile distances do not depend strongly on a few noisy samples around the decision boundary and, hence, the model is successfully able to resist the effect of noisy samples on classification accuracy. Though the Pin-SVM is noise insensitive, it still suffers from a high time complexity and, as such, is only useful for small to medium sized datasets. This project combines the properties of Pin-SVM and puts them in a twin support vector machine formulation by proposing a novel Sparse Pinball Twin Support Vector Machine (SPTWSVM). In order to further improve our model's capabilities, we then put forth a novel Improved Sparse Pinball Twin Support Vector Machine (ISPTWSVM) which works for large scale datasets by not having to calculate inverse of matrices that may be too large or singular.
Major contributions of the two-pronged project are summarized below:

- SPTWSVM
  - SPTWSVM is insensitive to outliers, retains the sparsity of solution and is stable for re-sampling as compared to TWSVM.
  - Since SPTWSVM solves two smaller QPPs rather than solving a single large QPP in case of Sparse Pin SVM, its time complexity is approximately four times faster and, thus performs predictions in remarkably less computational time.
  - Numerical results obtained when the SPTWSVM is applied on benchmark and artificial datasets demonstrate that the classification accuracy of the proposed model i.e. SPTWSVM outperforms the classification accuracy of the TWSVM and Sparse Pin SVM in most cases.
  - In [31], Xu et al. introduced the pinball loss into TPMSVM which is an extension of TWSVM. On the other hand, in this paper we introduce noise insensitivity and sparsity into the original TWSVM. Therefore, SPTWSVM can be easily extended to models which are built upon TWSVM.
- ISPTWSVM
  - ISPTWSVM, obtained by introducing changes in the primal form of TBSVM, is feasible for application on real world large scale datasets.
  - ISPTWSVM minimizes the structural risk in its formulation unlike SPTWSVM, which minimizes empirical risk instead. This embodies the marrow of statistical learning theory, and, consequently, classification accuracy on datasets can be improved due to this change.

– ISPTWSVM becomes insensitive to outliers and retains sparsity, achieved by the introduction of sparse pinball loss to the changed TBSVM problem.

Chapter 4

Design Proposal

Here we describe the formulations of our works in detail. Both SPTWSVM and ISPTWSVM have been described for the linear and non-linear cases. Steps have been properly highlighted with significance of specific terms explained in context.

4.1 Proposed Sparse Pinball Twin Support Vector Machines (SPTWSVM)

We combine the noise insensitivity and sparsity of Sparse Pin SVM and the speedup of TWSVM into our SPTWSVM formulation which is based on the ϵ-insensitive pinball loss function (1). The pinball loss function (and the ϵ-insensitive pinball loss by extension) can be considered to minimize the margin between the lower $\frac{\tau}{\tau+1}$ quantiles of the data samples. This property leads to the SVM model using pinball loss or its ϵ-insensitive variant being robust to feature noise in the data samples.

4.1.1 Linear Case

Following the method of formulating the first TWSVM problem (2a), we incorporate the sparse pinball loss function in the objective function to get the problem,

$$\min_{w^{(1)},b^{(1)}} \quad \frac{1}{2}(Aw^{(1)} + e_1 b^{(1)})^T(Aw^{(1)} + e_1 b^{(1)}) \; + \; c_1 e_2^T L_\tau^\epsilon(e_2 + (Bw^{(1)} + e_2 b^{(1)})) \tag{4}$$

and

$$\min_{w^{(2)},b^{(2)}} \quad \frac{1}{2}(Bw^{(2)} + e_2 b^{(2)})^T(Bw^{(2)} + e_2 b^{(2)}) \; + \; c_2 e_1^T L_\tau^\epsilon(e_1 - (Aw^{(2)} + e_1 b^{(2)})). \tag{5}$$

Here $c_1, c_2 > 0$ and e_1, e_2 are vectors of dimensions m_1 and m_2 respectively. For further clarity, L_τ^ϵ:$\mathbb{R}^x \rightarrow \mathbb{R}^x$ is the multi-variate version of (1), where x is the dimension of input vector u. In both the problems (4) and (5), the first term of the objective function corresponds to minimizing the sum of the squared distances of the samples of the concerned class from the hyperplane of that class. Meanwhile, the second term seeks to minimize the sum of errors that arise according to whether the samples of the other class are at least 1 unit distance away from the hyperplane or not. Problems (4) and (5) are converted into the equivalent familiar formulations of (2a) and

(2b) with the introduction of a slack vector ξ as follows:

$$\begin{aligned} \min_{w^{(1)},b^{(1)},\xi} \quad & \frac{1}{2}(Aw^{(1)} + e_1 b^{(1)})^T (Aw^{(1)} + e_1 b^{(1)}) \; + \; c_1 e_2^T \xi \\ \text{subject to} \quad & -(Bw^{(1)} + e_2 b^{(1)}) + \xi + e_2 \epsilon \geq e_2, \\ & -(Bw^{(1)} + e_2 b^{(1)}) \leq e_2 + \frac{\xi}{\tau} + e_2 \frac{\epsilon}{\tau}, \\ \text{and} \quad & \xi \geq 0. \end{aligned} \tag{6}$$

The above problem is the primal of the Sparse Pinball Twin Support Vector Machine for the first class (SPTWSVM1). In a similar fashion, we get SPTWSVM2,

$$\begin{aligned} \min_{w^{(2)},b^{(2)},\xi} \quad & \frac{1}{2}(Bw^{(2)} + e_2 b^{(2)})^T (Bw^{(2)} + e_2 b^{(2)}) \; + \; c_2 e_1^T \xi \\ \text{subject to} \quad & (Aw^{(2)} + e_1 b^{(2)}) + \xi + e_1 \epsilon \geq e_1, \\ & (Aw^{(2)} + e_1 b^{(2)}) \leq e_1 + \frac{\xi}{\tau} + e_1 \frac{\epsilon}{\tau}, \\ \text{and} \quad & \xi \geq 0. \end{aligned} \tag{7}$$

Here, ξ is used to provide an upper bound to the loss term. The error function we use here is the ϵ-insensitive zone pinball loss (1) which has a positive value on either side of the origin, thus, explaining the constraints of both (6) and (7).

We notice that both (6) and (7) are QPPs of the same form as the original TWSVM problems, the only difference being the extra constraint due to the different loss functions we use. For each of the two problems, we can see that the objective function depends on the samples of the corresponding class whereas the constraints depend on the samples of the other class.

To solve problems (6) and (7), we convert them to the dual form. We consider (6) for this purpose and introduce its Lagrangian function:

$$\begin{aligned} \mathcal{L}(w^{(1)}, b^{(1)}, \epsilon, \alpha, \beta, \gamma) = & \frac{1}{2}(Aw^{(1)} + e_1 b^{(1)})^T (Aw^{(1)} + e_1 b^{(1)}) + c_1 e_2^T \xi \\ & - \alpha^T (-(Bw^{(1)} + e_2 b^{(1)}) + \xi + e_2(\epsilon - 1)) - \beta^T (\xi) \\ & - \gamma^T ((Bw^{(1)} + e_2 b^{(1)}) + e_2(1 + \frac{\epsilon}{\tau}) + \frac{\xi}{\tau}), \end{aligned} \tag{8}$$

where $\alpha \geq 0$, $\beta \geq 0$ and $\gamma \geq 0$ are the Lagrangian multipliers. Applying the Karush-Kuhn-Tucker (KKT) optimality conditions, we get:

$$\frac{\partial \mathcal{L}}{\partial w^{(1)}} = A^T (Aw^{(1)} + e_1 b^{(1)}) + B^T \alpha - B^T \gamma = 0, \tag{9}$$

$$\frac{\partial \mathcal{L}}{\partial b^{(1)}} = e_1^T (Aw^{(1)} + e_1 b^{(1)}) + e_2^T \alpha - e_2^T \gamma = 0, \tag{10}$$

$$\frac{\partial \mathcal{L}}{\partial \xi} = c_1 e_2 - \alpha - \beta - \frac{\gamma}{\tau} = 0, \tag{11}$$

$$\alpha^T (-(Bw^{(1)} + e_2 b^{(1)}) + \xi + e_2(\epsilon - 1)) = 0, \tag{12}$$

$$\beta^T \xi = 0, \tag{13}$$

$$\gamma^T ((Bw^{(1)} + e_2 b^{(1)}) + e_2(1 + \frac{\epsilon}{\tau}) + \frac{\xi}{\tau}) = 0. \tag{14}$$

We combine constraints (9) and (10) to get,

$$\begin{bmatrix} A^T \\ e_1^T \end{bmatrix} \begin{bmatrix} A & e_1 \end{bmatrix} \begin{bmatrix} w^{(1)} \\ b^{(1)} \end{bmatrix} + \begin{bmatrix} B^T \\ e_2^T \end{bmatrix} (\alpha - \gamma) = 0 \tag{15}$$

and make the following substitutions:

$$\alpha - \gamma = \lambda, \tag{16}$$
$$H = \begin{bmatrix} A & e_1 \end{bmatrix}, \tag{17}$$
$$G = \begin{bmatrix} B & e_2 \end{bmatrix}, \tag{18}$$
$$\text{and} \quad u = \begin{bmatrix} w^{(1)} \\ b^{(1)} \end{bmatrix}. \tag{19}$$

Using the above substitutions, equation (15) may be rewritten as

$$H^T H u + G^T \lambda = 0, \quad \text{i.e,} \quad u = -(H^T H)^{-1} G^T \lambda. \tag{20}$$

$H^T H$ is always positive semi-definite, however, there lies the possibility that it may not be well conditioned in some situations. To remediate this issue, we usually add a small regularization term δI, $\delta > 0$ to (20), which is consistent with the method in Ridge Regression approaches such as [33]. This approach leads to the modified equation:

$$u = -(H^T H + \delta I)^{-1} G^T \lambda. \tag{21}$$

We, however, continue to use (20) elsewhere in the paper with the understanding that, if need be, (21) may be used.

Using the KKT conditions (9)-(11) and (16)-(19), our Lagrangian is modified to yield the dual problem of (6):

$$\begin{aligned} \max_{\lambda,\alpha} \quad & -\frac{1}{2}\lambda^T G (H^T H)^{-1} G^T \lambda + \lambda^T e_2 (\frac{\epsilon}{\tau} + 1) - \alpha^T e_2 (\epsilon + \frac{\epsilon}{\tau}) \\ \text{subject to} \quad & c_1 e_2 - \alpha - \beta - \frac{\gamma}{\tau} = 0, \\ & \alpha \geq 0, \quad \beta \geq 0, \quad \gamma \geq 0. \end{aligned} \tag{22}$$

Since $\beta \geq 0$, the first condition can equivalently be stated as $\alpha + \frac{\gamma}{\tau} \leq c_1 e_2$. Also, using $\gamma = \alpha - \lambda$, (22) can be rewritten as:

$$\begin{aligned} \min_{\lambda,\alpha} \quad & \frac{1}{2}\lambda^T G (H^T H)^{-1} G^T \lambda - \lambda^T e_2 (\frac{\epsilon}{\tau} + 1) + \alpha^T e_2 (\epsilon + \frac{\epsilon}{\tau}) \\ \text{subject to} \quad & \alpha (1 + \frac{1}{\tau}) - \frac{\lambda}{\tau} \leq c_1 e_2, \\ & \alpha \geq 0, \quad \alpha - \lambda \geq 0. \end{aligned} \tag{23}$$

Similarly, we can get the dual problem of (7) as follows:

$$\begin{aligned} \min_{\mu,\omega} \quad & \frac{1}{2}\mu^T P (Q^T Q)^{-1} P^T \mu - \mu^T e_1 (\frac{\epsilon}{\tau} + 1) + \omega^T e_1 (\epsilon + \frac{\epsilon}{\tau}) \\ \text{subject to} \quad & \omega (1 + \frac{1}{\tau}) - \frac{\mu}{\tau} \leq c_2 e_1, \\ & \omega \geq 0, \quad \omega - \mu \geq 0. \end{aligned} \tag{24}$$

Here, $P = \begin{bmatrix} A & e_1 \end{bmatrix}$, $Q = \begin{bmatrix} B & e_2 \end{bmatrix}$, and $\mu \geq 0$, $\omega \geq 0$ are Lagrangian multipliers. The vector $v = \begin{bmatrix} w^{(2)} \\ b^{(2)} \end{bmatrix}$ can be calculated, in a similar fashion, by the equation:

$$v = (Q^T Q)^{-1} P^T \mu \quad \text{or the well conditioned} \quad v = (Q^T Q + \delta I)^{-1} P^T \mu. \tag{25}$$

One can obtain the solutions of problems (23) and (24) and, subsequently, get the vectors u and v using which the two non-parallel hyperplanes can be defined:

$$x^T w^{(1)} + b^{(1)} = 0 \quad \text{and} \quad x^T w^{(2)} + b^{(2)} = 0. \tag{26}$$

Finally, the decision function to make a prediction for a new sample $x \in \mathbb{R}^n$ works by assigning the sample to class l, $(l = 1,\ 2)$ as follows:

$$l = \arg\min_{i=1,2} |x^T w^{(i)} + b^{(i)}|. \tag{27}$$

Here, $|.|$ is the perpendicular distance of data sample x from a given hyperplane.

4.1.2 Non-Linear Case

By employing the kernel trick, we extend our novel SPTWSVM to the non-linear case. Similar to Jayadeva et al. [15], we consider the non-linear surfaces:

$$\begin{aligned} K(x^T, C^T) z^{(1)} + b^{(1)} &= 0, \\ \text{and} \quad K(x^T, C^T) z^{(2)} + b^{(2)} &= 0, \\ \text{where} \quad C &= \begin{bmatrix} A_{m_1 \times n} \\ B_{m_2 \times n} \end{bmatrix}. \end{aligned} \tag{28}$$

Here, K is the kernel function which can be chosen according to the specific task at hand and $z^{(1)}, z^{(2)} \in \mathbb{R}^{(m_1+m_2)}$. For instance, if we choose the linear kernel then $K(x^T, C^T) = x^T C^T$ and define $C^T z^{(1)} = w^{(1)}$ and $C^T z^{(2)} = w^{(2)}$, then we get the linear planes in (26).

In a fashion similar to (6) and (7), we formulate the corresponding problems for the non-linear case:

$$\begin{aligned} \min_{z^{(1)}, b^{(1)}, \xi} \quad & \frac{1}{2} ||K(A, C^T) z^{(1)} + e_1 b^{(1)})||^2 + c_1 e_2^T \xi \\ \text{subject to} \quad & -(K(B, C^T) z^{(1)} + e_2 b^{(1)}) + \xi + e_2 \epsilon \ \geq \ e_2, \\ & -(K(B, C^T) z^{(1)} + e_2 b^{(1)}) \ \leq \ e_2 + \frac{\xi}{\tau} + e_2 \frac{\epsilon}{\tau}, \\ \text{and} \quad & \xi \ \geq \ 0. \end{aligned} \tag{29}$$

The above problem is the primal of the non-linear SPTWSVM1. Similarly, we get the primal of the non-linear SPTWSVM2,

$$\begin{aligned} \min_{z^{(2)}, b^{(2)}, \xi} \quad & \frac{1}{2} ||K(B, C^T) z^{(2)} + e_2 b^{(2)})||^2 + c_2 e_1^T \xi \\ \text{subject to} \quad & (K(A, C^T) z^{(2)} + e_1 b^{(2)}) + \xi + e_1 \epsilon \ \geq \ e_1, \\ & (K(A, C^T) z^{(2)} + e_1 b^{(2)}) \ \leq \ e_1 + \frac{\xi}{\tau} + e_1 \frac{\epsilon}{\tau}, \\ \text{and} \quad & \xi \ \geq \ 0. \end{aligned} \tag{30}$$

Here, $c_1 > 0,\ c_2 > 0$, and $e_1,\ e_2$ are vectors of ones of appropriate dimensions. We now consider problem (29) and derive its dual problem:

$$\begin{aligned} \min_{\lambda,\alpha} \quad & \frac{1}{2}\lambda^T R(S^T S)^{-1} R^T \lambda - \lambda^T e_2(\frac{\epsilon}{\tau} + 1) + \alpha^T e_2(\epsilon + \frac{\epsilon}{\tau}) \\ \text{subject to} \quad & \alpha(1 + \frac{1}{\tau}) - \frac{\lambda}{\tau} \ \leq \ c_1 e_2, \\ & \alpha \ \geq \ 0, \quad \alpha - \lambda \ \geq \ 0. \end{aligned} \tag{31}$$

Here, $S = \begin{bmatrix} K(A, C^T) & e_1 \end{bmatrix}$ and $R = \begin{bmatrix} K(B, C^T) & e_2 \end{bmatrix}$. The augmented vector $u = \begin{bmatrix} z^{(1)} \\ b^{(1)} \end{bmatrix}$ can be calculated as done in (20) by the relation:

$$u = -(S^T S) R^T \lambda. \tag{32}$$

We note that we apply well-conditioning, when required, in the same manner as in (21). Similarly, the dual problem for (30) is:

$$\begin{aligned} \min_{\mu,\omega} \quad & \frac{1}{2}\mu^T L(N^T N)^{-1} L^T \mu - \mu^T e_1(\frac{\epsilon}{\tau} + 1) + \omega^T e_1(\epsilon + \frac{\epsilon}{\tau}) \\ \text{subject to} \quad & \omega(1 + \frac{1}{\tau}) - \frac{\mu}{\tau} \ \leq \ c_2 e_1, \\ & \omega \ \geq \ 0, \quad \omega - \mu \ \geq \ 0. \end{aligned} \tag{33}$$

Here, $L = \begin{bmatrix} K(A, C^T) & e_1 \end{bmatrix}$ and $N = \begin{bmatrix} K(B, C^T) & e_2 \end{bmatrix}$. Further, the augmented vector $v = \begin{bmatrix} z^{(2)} \\ b^{(2)} \end{bmatrix}$ is calculated by the relation:

$$v = (N^T N) L^T \lambda. \tag{34}$$

Once we obtain the required parameters from problems (31) and (33), we use the decision function to predict the class of a new sample $x \in \mathbb{R}^n$ by assigning it to class $l,\ (l = 1\ ,\ 2)$ in a manner similar to the linear case.

4.2 Proposed Improved SPTWSVM for Large Scale Problems (ISPTWSVM)

In order to make our model's formulation suitable for large scale datasets, we introduce changes in the objective function of SPTWSVM which allow the dual problems of ISPTWSVM to bypass the calculation of large inverse matrices unlike the dual SPTWSVM problems. These changes are the introduction of a regularization term (as in TBSVM) and the addition of an equality constraint. Furthermore, since sparse pinball loss is already present in the primal problem of SPTWSVM, ISPTWSVM is noise insensitive and retains sparsity of the solution. ISPTWSVM also allows for the kernel trick to be incorporated directly into the dual problem instead of dealing with kernel generated surfaces, which is the case in SPTWSVM. Lastly, ISPTWSVM possesses the structural risk minimization principle unlike SPTWSVM, which gives ISPTWSVM the possibility of obtaining better classification accuracies on datasets. Thus, ISPTWSVM stands as a significant improvement over SPTWSVM and TBSVM.

4.2.1 Linear ISPTWSVM

Following the method of formulating the first TBSVM (3a) and SPTWSVM problem (4), we incorporate the sparse pinball loss function in the objective function to get the problems,

$$\begin{aligned} \min_{w^{(1)},b^{(1)},\eta_1,\xi} \quad & \frac{1}{2}c_3(||w^{(1)}||^2 + b^{(1)2}) + \frac{1}{2}\eta_1^T\eta_1 + c_1e_2^T\xi \\ \text{subject to} \quad & Aw^{(1)} + e_1b^{(1)} = \eta_1, \\ & -(Bw^{(1)} + e_2b^{(1)}) + \xi \ + e_2\epsilon \geq \ e_2, \\ & -(Bw^{(1)} + e_2b^{(1)}) \leq \ e_2 + \frac{\xi}{\tau} + e_2\frac{\epsilon}{\tau}, \\ \text{and} \quad & \xi \ \geq \ 0, \end{aligned} \tag{35}$$

and

$$\begin{aligned} \min_{w^{(2)},b^{(2)},\eta_2,\xi} \quad & \frac{1}{2}c_4(||w^{(2)}||^2 + b^{(2)2}) + \frac{1}{2}\eta_2^T\eta_2 + c_2e_1^T\xi \\ \text{subject to} \quad & Bw^{(2)} + e_2b^{(2)} = \eta_2, \\ & (Aw^{(2)} + e_1b^{(2)}) + \xi + e_1\epsilon \geq e_1, \\ & (Aw^{(2)} + e_1b^{(2)}) \leq e_1 + \frac{\xi}{\tau} + e_1\frac{\epsilon}{\tau}, \\ \text{and} \quad & \xi \geq 0. \end{aligned} \tag{36}$$

Here c_1, c_2, c_3, $c_4 > 0$, $\eta_1 \in \mathbb{R}^{m_1}$, $\eta_2 \in \mathbb{R}^{m_2}$, ξ is a slack vector which places an upper bound on the error terms, and e_1 and e_2 are vectors of ones with m_1 and m_2 elements respectively. Just like SPTWSVM, the third terms in both problems seek to minimize the sum of errors that arise according to whether the samples of the other class are at least 1 unit distance away from the hyperplane or not. The error function we use here is the ϵ-insensitive zone pinball loss (1) which has a positive value on either side of the origin, thus, explaining the constraints of both (35) and (36).

We notice that both (35) and (36) are QPPs of the same form as the original SPTWSVM problems, the only difference being the introduction of the regularization terms $\frac{1}{2}c_3(||w^{(1)}||^2 + b^{(1)2})$ and $\frac{1}{2}c_4(||w^{(2)}||^2 + b^{(2)2})$ and the addition of one extra equality constraint in both primal problems. The addition of the regularization terms also introduces structural risk minimization since they correspond to the distance between the proximal hyperplane, $w^{(1)T}x + b^{(1)} = 0$, and the bounding hyperplane, $w^{(1)T}x + b^{(1)} = -1$ (both planes correspond to the first problem). For each of the two problems, we can see that the objective function depends on samples of both classes; on one hand we wish to minimize the distances of the samples of a given class from the problem's corresponding hyperplane, while on the other hand we wish to minimize the error term associated with the samples of the other class.

To solve problems (35) and (36), we convert them to the dual form. We consider (35) for this purpose and introduce its Lagrangian function:

$$\begin{aligned} \mathcal{L}(w^{(1)}, b^{(1)}, \eta_1, \xi, \epsilon, \alpha, \beta, \gamma, \mu) = & \frac{1}{2}c_3(||w^{(1)}||^2 + b^{(1)2}) + \frac{1}{2}\eta_1^T\eta_1 + c_1e_2^T\xi \\ & - \alpha^T(-(Bw^{(1)} + e_2b^{(1)}) + \xi + e_2(\epsilon - 1)) - \beta^T(\xi) \\ & - \gamma^T((Bw^{(1)} + e_2b^{(1)}) + e_2(1 + \frac{\epsilon}{\tau}) + \frac{\xi}{\tau}) \\ & + \mu^T(Aw^{(1)} + e_1b^{(1)} - \eta_1), \end{aligned} \tag{37}$$

where $\alpha \geq 0$, $\beta \geq 0$, $\gamma \geq 0$ and $\mu \in \mathbb{R}^{m_1}$ are the Lagrangian multipliers. Applying the Karush-Kuhn-Tucker (KKT) optimality conditions, we get:

$$\frac{\partial \mathcal{L}}{\partial w^{(1)}} = c_3 w^{(1)} + B^T\alpha - B^T\gamma + A^T\mu = 0, \tag{38}$$

$$\frac{\partial \mathcal{L}}{\partial b^{(1)}} = c_3 b^{(1)} + e_2^T\alpha - e_2^T\gamma + e_1^T\mu = 0, \tag{39}$$

$$\frac{\partial \mathcal{L}}{\partial \xi} = c_1 e_2 - \alpha - \beta - \frac{\gamma}{\tau} = 0, \tag{40}$$

$$\frac{\partial \mathcal{L}}{\partial \eta_1} = \eta_1 - \mu = 0, \tag{41}$$

$$\alpha^T(-(Bw^{(1)} + e_2 b^{(1)}) + \xi + e_2(\epsilon - 1)) = 0, \tag{42}$$

$$\beta^T\xi = 0, \tag{43}$$

$$\gamma^T((Bw^{(1)} + e_2 b^{(1)}) + e_2(1 + \frac{\epsilon}{\tau}) + \frac{\xi}{\tau}) = 0, \tag{44}$$

$$\mu^T(Aw^{(1)} + e_1 b^{(1)} - \eta_1) = 0. \tag{45}$$

Using the KKT conditions (38)-(41) and (42)-(45) and substituting α - γ = λ, our Lagrangian is modified to yield the dual problem of (35):

$$\begin{aligned} \max_{\lambda,\alpha,\mu} \quad & -\frac{1}{2}[\mu^T\lambda^T]\tilde{Q}\begin{bmatrix}\mu\\ \lambda\end{bmatrix} + c_3\lambda^T e_2(\frac{\epsilon}{\tau} + 1) - c_3\alpha^T e_2(\epsilon + \frac{\epsilon}{\tau}) \\ \text{subject to} \quad & c_1 e_2 - \alpha - \beta - \frac{\gamma}{\tau} = 0, \\ & \alpha \geq 0, \quad \beta \geq 0, \quad \gamma \geq 0, \\ \text{where} \quad & \tilde{Q} = \begin{bmatrix} A^TA + c_3 I & AB^T \\ BA^T & BB^T \end{bmatrix} + E. \end{aligned} \tag{46}$$

Here, E is a matrix of all ones of size $(m_1 + m_2) \times (m_1 + m_2)$. Since $\beta \geq 0$, the first condition can equivalently be stated as $\alpha + \frac{\gamma}{\tau} \leq c_1 e_2$. Also, using $\gamma = \alpha - \lambda$, (46) can be rewritten as:

$$\begin{aligned} \min_{\mu,\lambda,\alpha} \quad & \frac{1}{2}[\mu^T\lambda^T]\tilde{Q}\begin{bmatrix}\mu\\ \lambda\end{bmatrix} - c_3\lambda^T e_2(\frac{\epsilon}{\tau} + 1) + c_3\alpha^T e_2(\epsilon + \frac{\epsilon}{\tau}) \\ \text{subject to} \quad & \alpha(1 + \frac{1}{\tau}) - \frac{\lambda}{\tau} \leq c_1 e_2, \\ & \alpha \geq 0, \quad \alpha - \lambda \geq 0, \\ \text{where} \quad & \tilde{Q} = \begin{bmatrix} A^TA + c_3 I & AB^T \\ BA^T & BB^T \end{bmatrix} + E. \end{aligned} \tag{47}$$

Similarly, we can get the dual problem of (36) as follows:

$$\begin{aligned} \min_{\theta,\phi,\omega} \quad & \frac{1}{2}[\theta^T\phi^T]\tilde{Q}\begin{bmatrix}\theta\\ \phi\end{bmatrix} - c_4\phi^T e_1(\frac{\epsilon}{\tau} + 1) + c_4\omega^T e_1(\epsilon + \frac{\epsilon}{\tau}) \\ \text{subject to} \quad & \omega(1 + \frac{1}{\tau}) - \frac{\phi}{\tau} \leq c_2 e_1, \\ & \omega \geq 0, \quad \omega - \phi \geq 0, \\ \text{where} \quad & \tilde{Q} = \begin{bmatrix} B^TB + c_4 I & -BA^T \\ -AB^T & AA^T \end{bmatrix} + E. \end{aligned} \tag{48}$$

One can obtain the solutions of problems (47) and (48) and, subsequently, get the vectors $\begin{bmatrix}\mu & \lambda & \alpha\end{bmatrix}$ and $\begin{bmatrix}\theta & \phi & \omega\end{bmatrix}$ using which the two non-parallel hyperplanes can be defined:

$$x^T w^{(1)} + b^{(1)} = 0 \quad \text{and} \quad x^T w^{(2)} + b^{(2)} = 0. \tag{49}$$

Finally, the decision function to make a prediction for a new sample $x \in \mathbb{R}^n$ is similar to the SPTWSVM problem.

4.2.2 Non-Linear ISPTWSVM

Unlike the SPTWSVM non-linear case, we need not consider kernel generated surfaces for ISPTWSVM and can directly introduce the kernel function in the linear case of ISPTWSVM. Hence, we introduce the kernel function $K(x,y) = \phi(x)^T\phi(y)$ into the linear case, where we have the transformation $\mathbf{x} = \phi(x)$, $\mathbf{x} \in \mathbb{H}$ (Hilbert space). In a similar fashion to (35) and (36), we now consider the following primal problems in the Hilbert space $\mathbb{H}$:

$$\begin{aligned}
\min_{w^{(1)},b^{(1)},\eta_1,\xi} \quad & \frac{1}{2}c_3(||w^{(1)}||^2 + b^{(1)2}) + \frac{1}{2}\eta_1^T\eta_1 + c_1 e_2^T\xi \\
\text{subject to} \quad & \phi(A)w^{(1)} + e_1 b^{(1)} = \eta_1, \\
& -(\phi(B)w^{(1)} + e_2 b^{(1)}) + \xi \ + e_2\epsilon \geq \ e_2, \\
& -(\phi(B)w^{(1)} + e_2 b^{(1)}) \leq \ e_2 + \frac{\xi}{\tau} + e_2\frac{\epsilon}{\tau}, \\
\text{and} \quad & \xi \ \geq \ 0,
\end{aligned} \tag{50}$$

and

$$\begin{aligned}
\min_{w^{(2)},b^{(2)},\eta_2,\xi} \quad & \frac{1}{2}c_4(||w^{(2)}||^2 + b^{(2)2}) + \frac{1}{2}\eta_2^T\eta_2 + c_2 e_1^T\xi \\
\text{subject to} \quad & \phi(B)w^{(2)} + e_2 b^{(2)} = \eta_2, \\
& (\phi(A)w^{(2)} + e_1 b^{(2)}) + \xi + e_1\epsilon \geq e_1, \\
& (\phi(A)w^{(2)} + e_1 b^{(2)}) \leq e_1 + \frac{\xi}{\tau} + e_1\frac{\epsilon}{\tau}, \\
\text{and} \quad & \xi \geq 0.
\end{aligned} \tag{51}$$

Here, all constants and notations have the same meaning from the linear case. We derive the dual problem of (50) and (51):

$$\begin{aligned}
\min_{\mu,\lambda,\alpha} \quad & \frac{1}{2}[\mu^T\lambda^T]\tilde{Q}\begin{bmatrix}\mu\\ \lambda\end{bmatrix} - c_3\lambda^T e_2(\frac{\epsilon}{\tau} + 1) + c_3\alpha^T e_2(\epsilon + \frac{\epsilon}{\tau}) \\
\text{subject to} \quad & \alpha(1 + \frac{1}{\tau}) - \frac{\lambda}{\tau} \leq c_1 e_2, \\
& \alpha \geq 0, \quad \alpha - \lambda \geq 0, \\
\text{where} \quad & \tilde{Q} = \begin{bmatrix}K(A^T, A^T) + c_3 I & K(A^T, B^T) \\ K(B^T, A^T) & K(B^T, B^T)\end{bmatrix} + E,
\end{aligned} \tag{52}$$

and

$$\begin{aligned} \min_{\theta,\phi,\omega} \quad & \frac{1}{2}[\theta^T \phi^T]\tilde{Q}\begin{bmatrix}\theta\\ \phi\end{bmatrix} - c_4\phi^T e_1(\frac{\epsilon}{\tau}+1) + c_4\omega^T e_1(\epsilon+\frac{\epsilon}{\tau}) \\ \text{subject to} \quad & \omega(1+\frac{1}{\tau}) - \frac{\phi}{\tau} \leq c_2 e_1, \\ & \omega \geq 0, \quad \omega - \phi \geq 0, \\ \text{where} \quad & \tilde{Q} = \begin{bmatrix} K(B^T, B^T) + c_4 I & -K(B^T, A^T) \\ -K(A^T, B^T) & K(A^T, A^T) \end{bmatrix} + E. \end{aligned} \tag{53}$$

All variables, constants and notations are similar to those from the linear case. Once we obtain the required parameters from problems (52) and (53), we use the decision function to predict the class of a new sample $x \in \mathbb{R}^n$ by assigning it to class l, $(l = 1\ ,\ 2)$ in a manner similar to the linear case.

4.3 Performance Evaluation Metrics

All experiments in this project deal with demonstrating noise insensitivity and resampling stability in both UCI and synthetic datasets, sparsity in UCI datasets, and general classification accuracy in UCI datasets.

Chapter 5

Novel SPTWSVM: Properties and its Analytical Arguments

In this section, we discuss some properties and results based on our SPTWSVM .

5.1 Noise Insensitivity

Here we explain, from an analytical perspective, how incorporating the ϵ-insensitive pinball function leads to noise insensitivity. For the sake of brevity, we consider SPTWSVM1 (23) for the linear case (the same analysis applies to SPTWSVM1 for the non-linear case and SPTWSVM2 for both the linear and non-linear cases). Consider the generalized sign function, $\text{sgn}_\tau^\epsilon(x)$, corresponding to (1):

$$\text{sgn}_\tau^\epsilon(x) = \begin{cases} 1, & x > \epsilon, \\ [0,1], & x = \epsilon, \\ 0, & -\frac{\epsilon}{\tau} < x < \epsilon, \\ [-\tau, 0], & x = \frac{-\epsilon}{\tau}, \\ -\tau, & x < -\frac{\epsilon}{\tau}. \end{cases} \tag{54}$$

$\text{sgn}_\tau^\epsilon(x)$ is the subgradient of the ϵ-insensitive pinball loss function and, hence, the optimality condition for (4) can be written as:

$$\mathbf{0} \in A^T(Aw^{(1)} + e_1 b^{(1)}) + c_1 \sum_{i=1}^{m_2} \text{sgn}_\tau^\epsilon(1 + (w^{(1)T} x_i^- + b^{(1)}))x_i^-, \tag{55}$$

where $\mathbf{0}$ is the vector which has all its components equal to zero and $x_i^- \in B$.

For a given $w^{(1)}, b^{(1)}$, we partition the index set into five sets:

$$\begin{aligned} S_0^{w^{(1)},b^{(1)}} &= \{i : 1 + (w^{(1)T} x_i^- + b^{(1)}) > \epsilon\}, \\ S_1^{w^{(1)},b^{(1)}} &= \{i : 1 + (w^{(1)T} x_i^- + b^{(1)}) = \epsilon\}, \\ S_2^{w^{(1)},b^{(1)}} &= \{i : \frac{-\epsilon}{\tau} < 1 + (w^{(1)T} x_i^- + b^{(1)}) < \epsilon\}, \\ S_3^{w^{(1)},b^{(1)}} &= \{i : 1 + (w^{(1)T} x_i^- + b^{(1)}) = \frac{-\epsilon}{\tau}\}, \\ S_4^{w^{(1)},b^{(1)}} &= \{i : 1 + (w^{(1)T} x_i^- + b^{(1)}) < \frac{-\epsilon}{\tau}\}. \end{aligned}$$

Here, $i \in \{1, 2, ..., m_2\}$. The data samples in $S_2^{w^{(1)},b^{(1)}}$ do not contribute to $w^{(1)}$ since the subgradient at these data samples is zero, as is evident from (53). Thus, $S_2^{w^{(1)},b^{(1)}}$ directly affects sparsity of the model. Set $S_2^{w^{(1)},b^{(1)}}$ is dependent on the value of ϵ. As ϵ approaches 0 sparsity is

lost whereas if $\epsilon \to \infty$, more samples lie in $S_2^{w^{(1)},b^{(1)}}$ and, as a result, we gain sparsity. With the above notations and the existence of $\psi_i \in [0,1]$ and $\theta_i \in [-\tau, 0]$ equation (54) can be rewritten as:

$$\frac{1}{c_1} A^T(Aw^{(1)} + e_1 b^{(1)}) + \sum_{i \in S_0^{w^{(1)},b^{(1)}}} x_i^- + \sum_{i \in S_1^{w^{(1)},b^{(1)}}} \psi_i x_i^- + \sum_{i \in S_3^{w^{(1)},b^{(1)}}} \theta_i x_i^- - \tau \sum_{i \in S_4^{w^{(1)},b^{(1)}}} x_i^- = \mathbf{0}. \tag{56}$$

The above condition shows that when the value of ϵ is fixed, τ controls the number of samples in the sets $S_0^{w^{(1)},b^{(1)}}$, $S_1^{w^{(1)},b^{(1)}}$, $S_2^{w^{(1)},b^{(1)}}$, $S_3^{w^{(1)},b^{(1)}}$, and $S_4^{w^{(1)},b^{(1)}}$. However, since the number of data samples in $S_1^{w^{(1)},b^{(1)}}$ and $S_3^{w^{(1)},b^{(1)}}$ are much fewer than in the other sets, we are primarily concerned with sets $S_0^{w^{(1)},b^{(1)}}$, $S_2^{w^{(1)},b^{(1)}}$ and $S_4^{w^{(1)},b^{(1)}}$. When τ is small, the number of samples in $S_4^{w^{(1)},b^{(1)}}$ is quite large while the other sets have fewer data samples, thus making the result sensitive to feature noise in the samples. On the contrary, having a larger τ value imparts many data samples to all the five sets and the result is less sensitive to zero mean feature noise.

Proposition 1. *If the optimization problem (23) or (31) has a solution then the following inequalities must hold:*

$$\frac{A^T(Aw^{(1)} + e_1 b^{(1)})}{c_1 m_2} \leq 1 \text{ and } \frac{p_0}{m_2} \leq 1 - \frac{1 - \frac{A^T(Aw^{(1)} + e_1 b^{(1)})}{c_1 m_2}}{1 + \tau},$$

where p_0 denotes the number of samples in $S_0^{w^{(1)},b^{(1)}}$.

Proof. Consider an arbitrary sample $x_{i_0}^- \in S_0^{w^{(1)},b^{(1)}}$, $(1 \leq i_0 \leq m_2)$. From the KKT conditions (13) and (14), $\beta_{i_0} = \gamma_{i_0} = 0$. From the KKT condition (11), we then obtain $\alpha_{i_0} = c_1$ and, subsequently, $\lambda_{i_0} = \alpha_{i_0} - \gamma_{i_0} = c_1$. Also, from the KKT condition (10), we have

$$\sum_{i \in S_0^{w^{(1)},b^{(1)}}} \lambda_i + \sum_{i \notin S_0^{w^{(1)},b^{(1)}}} \lambda_i = e_1^T(Aw^{(1)} + e_1 b^{(1)}) \implies p_0 c_1 + \sum_{i \notin S_0^{w^{(1)},b^{(1)}}} \lambda_i = e_1^T(Aw^{(1)} + e_1 b^{(1)}).$$

Now, since $\alpha_i \geq 0$ and $\gamma_i \geq 0$, we have $-\tau c_1 \leq \lambda_i \leq c_1$. Therefore,

$$\frac{e_1^T(Aw^{(1)} + e_1 b^{(1)})}{c_1} - (m_2 - p_0) \leq p_0 \leq \frac{e_1^T(Aw^{(1)} + e_1 b^{(1)})}{c_1} + \tau(m_2 - p_0),$$

which gives us $\frac{e_1^T(Aw^{(1)} + e_1 b^{(1)})}{c_1 m_2} \leq 1$ and $p_0(1+\tau) \leq \frac{e_1^T(Aw^{(1)} + e_1 b^{(1)}) + \tau c_1 m_2}{c_1}$. The second condition gives us

$$\frac{p_0}{m_2} \leq \frac{\frac{e_1^T(Aw^{(1)} + e_1 b^{(1)})}{m_2} + \tau c_1}{c_1(1+\tau)} = 1 - \frac{c_1 - \frac{e_1^T(Aw^{(1)} + e_1 b^{(1)})}{m_2}}{c_1(1+\tau)} = 1 - \frac{1 - \frac{e_1^T(Aw^{(1)} + e_1 b^{(1)})}{c_1 m_2}}{(1+\tau)}$$

hence proving our proposition. □

As is evident, the above proposition places an upper bound on the number of samples in $S_0^{w^{(1)},b^{(1)}}$; when τ becomes small, p_0 gets smaller and the result becomes more sensitive to feature noise since a lot fewer data samples are distributed in sets other than $S_4^{w^{(1)},b^{(1)}}$. As a result, feature noise around the decision boundary significantly affects classification results. A similar analysis holds for SPTWSVM2 problems (24) and (33).

5.2 Scatter Minimization

One can interpret the mechanism of our SPTWSVM model through scatter minimization as well. Let data samples in subset $Y_1^{w^{(1)},b^{(1)}} \subset A$ determine the hyperplane $x^T w^{(1)} + b^{(1)} = 0$ and samples in subset $Y_2^{w^{(1)},b^{(1)}} \subset S_2^{w^{(1)},b^{(1)}}$ and subset $Y_2^{w^{(2)},b^{(2)}} \subset S_2^{w^{(2)},b^{(2)}}$ determine the two hyperplanes $\mathcal{H}' : \{w^{(1)T} x_i^- + b^{(1)} + 1 = 0\}$ and $\mathcal{H}'' : \{w^{(2)T} x_j^+ + b^{(2)} - 1 = 0\}$, respectively, where $x_i^- \in B$ and $x_j^+ \in A$. We use the sum of distances from each sample x_i^- to a given sample $x_{i_2}^- \in Y_2^{w^{(1)},b^{(1)}}$ to determine the scatter. Then we let scatter of sample $x_i^- \in B$ around data sample $x_{i_2}^-$ be defined by:

$$\sum_{i=1}^{m_2} |w^{(1)T}(x_{i_2}^- - x_i^-)|.$$

Since $w^{(1)T} x_{i_2}^- + b^{(1)} + 1 = 0$, we have

$$\sum_{i=1}^{m_2} |w^{(1)T}(x_{i_2}^- - x_i^-)| = \sum_{i=1}^{m_2} | - 1 - (w^{(1)T} x_i^- + b^{(1)})| = \sum_{i=1}^{m_2} |1 + (w^{(1)T} x_i^- + b^{(1)})|.$$

Similarly, we consider the scatter of each sample $x_j^+ \in A$ from a given data sample $x_{j_1}^+ \in Y_1^{w^{(1)},b^{(1)}}$ as follows:

$$\sum_{j=1}^{m_1} |w^{(1)T}(x_{j_1}^+ - x_j^+)| = \sum_{j=1}^{m_1} | - (w^{(1)T} x_j^+ + b^{(1)})|,$$

since $w^{(1)T} x_{j_1}^+ + b^{(1)} = 0$. As the scatter turns out to be a positive value, we can consider the scatter as the sum of squares, i.e., $\sum_{j=1}^{m_1} (-(w^{(1)T} x_j^+ + b^{(1)}))^2$.

Now consider the following formulation,

$$\min_{w^{(1)},b^{(1)}} \quad \frac{1}{2} \sum_{j=1}^{m_1} (-(w^{(1)T} x_j^+ + b^{(1)}))^2 + c_{11} \sum_{i=1}^{m_2} |1 + (w^{(1)T} x_i^- + b^{(1)})|. \tag{57}$$

Here, $c_{11} > 0$ is a constant. The first term can be interpreted to minimize the scatter of $x_j^+ \in A$ around the hyperplane $x^T w^{(1)} + b^{(1)} = 0$. Meanwhile, the second term seeks to minimize the scatter of $x_i^- \in B$ around the hyperplane $\mathcal{H}'$, which minimizes the error values that arise according to how close the samples of B are to $\mathcal{H}'$. In problem (4), the first term of (57) is stated in its mathematically equivalent form whereas the second term of (57) is extended to L_τ^ϵ by introducing the following misclassification terms:

$$\begin{aligned}
c_{12} L_{hinge}(1 + (w^{(1)T} x_i^- + b^{(1)}) - \epsilon) &= max(0, 1 + (w^{(1)T} x_i^- + b^{(1)}) - \epsilon), \\
c_{13} L_{hinge}(1 + (w^{(1)T} x_i^- + b^{(1)})) &= max(0, 1 + (w^{(1)T} x_i^- + b^{(1)})), \\
c_{14} L_{hinge}(-1 - (w^{(1)T} x_i^- + b^{(1)})) &= max(0, -1 - (w^{(1)T} x_i^- + b^{(1)})), \\
c_{15} L_{hinge}(-1 - (w^{(1)T} x_i^- + b^{(1)}) - \frac{\epsilon}{\tau}) &= max(0, -1 - (w^{(1)T} x_i^- + b^{(1)}) - \frac{\epsilon}{\tau}).
\end{aligned}$$

Here $c_{12}, c_{13}, c_{14}, c_{15} > 0$ are constants and we obtain problem (4) with the conditions: $c_{11} + c_{12} + c_{13} = c_1$, $c_{11} + c_{13} = 0$, $c_{11} + c_{14} = 0$, and $c_{11} + c_{14} + c_{15} = \tau c_1$. From the last condition, $\tau = \frac{c_{15}}{c_1}$ which suggests the reasonable range of $\tau \geq 0$. A similar analysis holds for problem (5).

In conclusion, SPTWSVM minimization considers both within-class scatter of one class and misclassification error of the other (which is also a case of scatter minimization around $\mathcal{H}^{'}$) together. The SPTWSVM problem (4) is then considered a trade-off between small scatter and small misclassification.

Chapter 6

Experiments

6.1 Datasets

In this section, the performance of the algorithm is tested on several benchmark UCI datasets. These datasets were originally proposed for binary classification problems. Apart from this, synthetic datasets are also used for calculating the performance of the SPTWSVM.

6.2 Experimental Setup:

We apply our SPTWSVM model to an artificial dataset and 10 benchmark UCI datasets to exhibit the accuracy, noise insensitivity and sparsity of our model. All of the experiments have been performed on MATLAB R2017a on a Windows 10 machine with an Intel i5 Processor (3.4 GHz) with 16 GB RAM.

6.2.1 SPTWSVM

To solve our SPTWSVM model with lower computational complexity, we make $c_1 = c_2 = c$, $\tau_1 = \tau_2 = \tau$, and $\epsilon_1 = \epsilon_2 = \epsilon$. In all our experiments, c is chosen from the set $\{10^i : i = -5,\ -4,\ -3,\ ..., +3,\ +4,\ +5\}$, τ is chosen from the set $\{0.01,\ 0.1,\ 0.2,\ 0.5,\ 1\}$ and ϵ is chosen from the set $\{0,\ 0.05,\ 0.1,\ 0.2,\ 0.3,\ 0.5\}$.

6.2.2 ISPTWSVM

To solve our ISPTWSVM model with lower computational complexity, we make $c_1 = c_2 = c$ and $c_3 = c_4 = c'$, $\tau_1 = \tau_2 = \tau$, and $\epsilon_1 = \epsilon_2 = \epsilon$. In all our experiments, c and c' are chosen from the set $\{10^i : i = -5,\ -4,\ -3,\ ..., +3,\ +4,\ +5\}$, τ and ϵ are chosen similar to SPTWSVM.

6.3 Synthetic Dataset:

The purpose of our SPTWSVM is to be able to deal with noise around the decision boundary while retaining sparsity. To illustrate the noise insensitivity performance consider Fig. 6.1, where we take a two dimensional synthetic dataset with equal number of samples from two Gaussian distributions: $x_i, i \in \{i : y_i = 1\} \sim \mathcal{N}(\mu_1, \sum_1)$ and $x_i, i \in \{i : y_i = -1\} \sim \mathcal{N}(\mu_2, \sum_2)$ where $\mu_1 = [0.5, -3]^T, \mu_2 = [-0.5, 3]^T$ and $\sum_1 = \sum_2 = \begin{bmatrix} 0.2 & 0 \\ 0 & 3 \end{bmatrix}$. The Bayes classifier for the given Gaussian distribution is $f_c(x) = 2.5x(1) - x(2)$, that is, the ideal result is a separating hyperplane with slope equal to 2.5 and y-intercept equal to 0. We now add noise to the dataset, with each noise sample drawn from the Gaussian distribution $\mathcal{N}(\mu_n, \sum_n)$ where $\mu_n = [0, 0]^T$ and $\sum_n = \begin{bmatrix} 1 & -0.8 \\ -0.8 & 1 \end{bmatrix}$. Each noise sample is assigned the label +1 or -1 with equal probability. Also, the

number of noisy data samples is determined by r, the ratio of noisy samples to total samples in the original distribution (100 each of class +1 and -1).

The noise samples affect the labels around the decision boundary; however, the Bayes classifier for such a noise filled distribution still remains the same. In Fig. 6.1, we can see that as we increase the amount of noise (from $r = 0$ to $r = 0.2$), the hyperplanes of C-SVM and TWSVM start deviating from the ideal slope of 2.5 whereas the deviation in the slopes of hyperplanes (with fixed values of $\tau = 0.5$ and $\epsilon = 0.05$) is significantly lesser in our SPTWSVM. This implies the sensitivity of the TWSVM and C-SVM models to noise around the boundary.

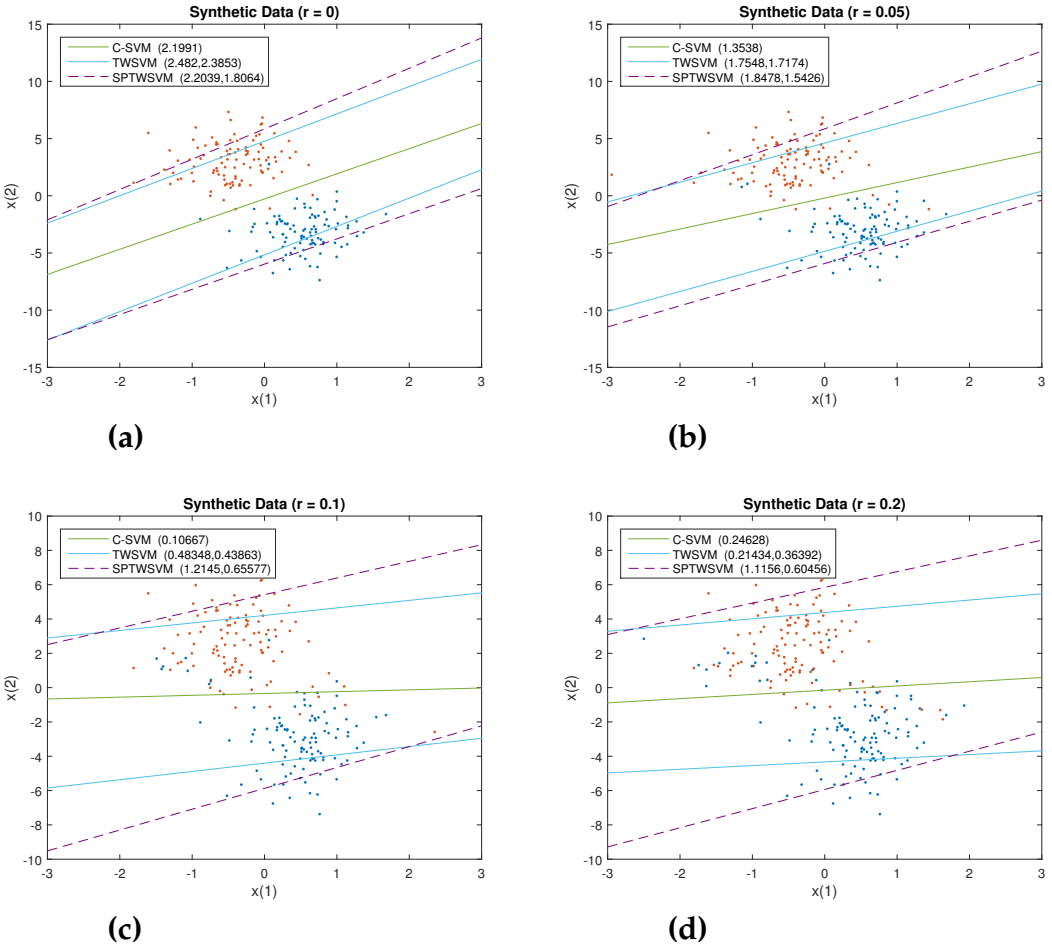


Figure 6.1: The above four figures demonstrate the noise insensitive properties possessed by our SPTWSVM as compared to C-SVM and TWSVM when we have varying number of noise samples, from $r = 0$ (noise free) to $r = 0.2$. Here, r is the ratio of total number of noisy samples to the total number of samples originally in the dataset (including both classes). The legend in each figure matches a given model to its corresponding hyperplane and gives the slopes of the separating hyperplanes in the brackets.

6.4 UCI Datasets

After noticing the noise insensitive performance of the SPTWSVM model on synthetic data, we consider ten real world datasets, downloaded from the UCI Repository of Machine Learning Dataset [34]. Wherever an explicit train-test split has not been provided, we have randomly partitioned the dataset into two equal train and test datasets.

Table 6.1 summarizes the results for a linear kernel, $K(x^T, C^T) = x^T C^T$, on six different UCI datasets. Here, we compare the accuracy of our novel SPTWSVM model with that of the Sparse Pin SVM, proposed by Huang et al. [16] and the TWSVM proposed by Jayadeva et al. [15]. The optimal value of c for TWSVM and for each (ϵ, τ) combination for Sparse Pin SVM and SPTWSVM is computed using ten-fold cross-validation. For each dataset, we apply Sparse Pin SVM, TWSVM and SPTWSVM to perform classification for different (ϵ, τ) combinations, and subsequently the model with the best classification accuracy is highlighted in bold. In the table, the total number of samples and the number of features have been highlighted below each dataset. From the results tabulated in Table 6.1, one can observe that classification performance of SPTWSVM is better than that of Sparse Pin SVM and TWSVM in most of the datasets. However, in Heart-C the classification accuracies of Sparse Pin SVM and TWSVM are better than that of SPTWSVM. This might be attributed to a different distribution of samples in the Heart-C dataset.

Table 6.1: Accuracy obtained on UCI datasets with a linear kernel for SPTWSVM

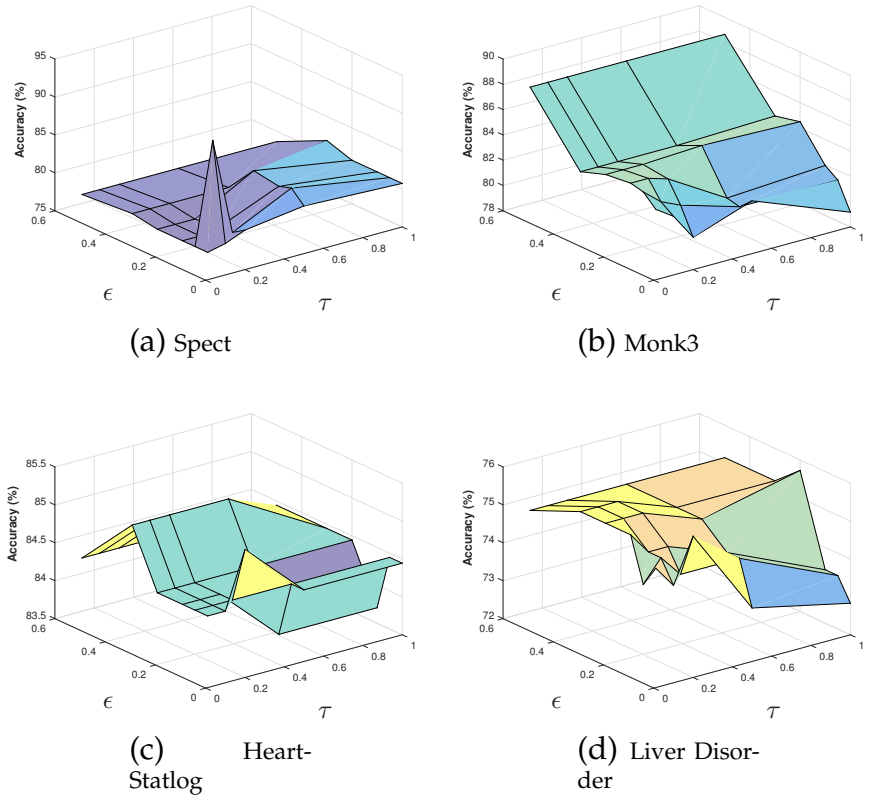
Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**SPTWSVM**				
		τ						τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Heart-Statlog	0	85.93	87.41	85.18	82.96	77.04	87.41	87.41	87.41	87.41	87.41	87.41
(270 × 13)	0.05	85.93	87.41	85.93	82.96	76.3	87.41	87.41	87.41	87.41	87.41	87.41
	0.1	85.2	**88.15**	85.19	83.7	77.04	87.41	87.41	87.41	87.41	**88.15**	87.41
	0.2	85.18	87.41	87.41	83.7	82.96	87.41	87.41	87.41	87.41	87.41	87.41
	0.3	85.18	88.15	**88.15**	85.18	83.7	87.41	87.41	87.41	87.41	87.41	87.41
	0.5	87.41	87.41	**88.15**	**88.15**	88.15	87.41	87.41	87.41	87.41	87.41	87.41
Australian	0	84.68	85.26	85.55	85.26	85.55	**86.71**	**86.71**	86.42	86.13	85.84	85.84
(690 × 14)	0.05	84.1	85.26	85.26	85.26	85.55	**86.71**	**86.71**	86.42	86.42	86.13	85.84
	0.1	84.1	85.26	85.26	85.26	85.26	**86.71**	**86.71**	86.42	86.42	86.13	85.84
	0.2	84.4	85.55	85.26	85.26	85.26	**86.71**	86.42	86.42	86.13	85.84	85.84
	0.3	84.97	85.55	85.26	85.26	85.55	**86.71**	85.84	85.84	85.84	85.84	85.84
	0.5	84.68	85.84	85.55	85.54	85.55	**86.71**	86.13	86.13	86.13	86.13	86.13
Heart-C	0	**82.24**	81.58	78.29	71.05	71.71	**82.24**	81.58	80.26	80.92	80.26	80.26
(303 × 13)	0.05	**82.24**	80.92	79.61	71.05	71.05	**82.24**	81.58	80.92	80.92	80.26	80.26
	0.1	**82.24**	80.92	78.29	71.05	71.05	**82.24**	81.58	81.58	81.58	80.26	80.26
	0.2	81.58	81.58	78.95	71.05	71.05	**82.24**	81.58	81.58	81.58	81.58	80.26
	0.3	81.58	81.58	79.61	73.03	73.03	**82.24**	80.92	80.92	80.92	80.92	81.58
	0.5	**82.24**	81.58	81.58	79.61	79.61	**82.24**	80.26	80.26	80.26	80.26	80.26
SPECT	0	73.8	74.33	75.4	72.73	72.73	78.61	78.61	79.14	80.21	81.28	80.75
(267 × 22)	0.05	74.33	74.33	74.33	72.73	72.73	78.61	78.61	**91.98**	79.14	82.89	80.75
	0.1	74.33	73.26	74.33	72.73	72.73	78.61	78.61	78.61	79.14	82.35	80.75
	0.2	74.33	74.33	74.87	72.73	72.73	78.61	78.61	78.61	78.61	82.88	80.75
	0.3	74.33	73.8	74.87	73.26	72.73	78.61	79.14	79.14	79.14	79.14	81.82
	0.5	73.8	73.8	74.87	75.4	74.87	78.61	78.61	78.61	78.61	78.61	78.61
Monks3	0	81.71	81.71	81.02	81.25	81.94	84.03	83.57	82.64	80.56	82.18	79.17
(432 × 6)	0.05	81.94	81.71	81.02	80.79	80.79	84.03	84.49	83.33	82.64	81.25	81.25
	0.1	81.71	81.71	81.94	80.79	81.02	84.03	84.72	84.72	84.26	81.48	81.94
	0.2	80.56	81.71	81.02	81.25	81.25	84.03	84.49	84.49	84.49	84.72	84.49
	0.3	81.25	81.71	81.94	81.94	81.94	84.03	83.79	83.79	83.79	83.79	84.03
	0.5	81.48	80.79	81.71	81.94	82.64	84.03	**88.66**	**88.66**	**88.66**	**88.66**	**88.66**
Breast	0	70.69	70.69	**72.41**	**72.41**	68.97	68.97	**72.41**	68.97	70.69	**72.41**	**72.41**
(116 × 10)	0.05	70.69	70.69	**72.41**	70.69	70.69	68.97	70.69	70.69	70.69	**72.41**	**72.41**
	0.1	70.69	70.69	**72.41**	70.69	70.69	68.97	70.69	**72.41**	70.69	70.69	70.69
	0.2	70.69	70.69	**72.41**	**72.41**	70.69	68.97	70.69	70.69	70.69	**72.41**	**72.41**
	0.3	70.69	70.69	**72.41**	**72.41**	70.69	68.97	70.69	70.69	70.69	**72.41**	**72.41**
	0.5	70.69	70.69	70.69	**72.41**	**72.41**	68.97	70.69	**72.41**	70.69	**72.41**	**72.41**

Table 6.2: Accuracy obtained on UCI datasets with a non-linear kernel for SPTWSVM

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**SPTWSVM**				
		τ						τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Heart-Statlog	0	83.70	84.44	82.22	82.96	80.74	84.44	84.44	84.44	**85.18**	84.44	84.44
(270 × 13)	0.05	82.22	84.44	82.96	82.96	80.00	84.44	84.44	84.44	84.44	84.44	84.44
	0.1	81.48	84.44	82.96	82.96	81.48	84.44	84.44	84.44	84.44	83.70	83.70
	0.2	82.22	**85.18**	82.96	82.96	83.70	84.44	84.44	84.44	84.44	84.44	84.44
	0.3	82.22	84.44	82.96	82.22	84.44	84.44	**85.18**	**85.18**	**85.18**	**85.18**	84.44
	0.5	83.70	84.44	**85.18**	83.70	83.70	84.44	84.44	84.44	84.44	84.44	84.44
Sonar	0	60.95	56.19	60.00	56.19	56.19	62.86	**70.48**	**70.48**	**70.48**	68.57	68.57
(208 × 60)	0.05	63.81	56.19	56.19	60.95	60.00	62.86	62.86	61.90	61.90	61.90	61.90
	0.1	62.86	60.95	65.71	63.81	61.90	62.86	61.90	61.90	61.90	61.90	61.90
	0.2	55.24	63.81	63.81	62.86	63.81	62.86	61.90	61.90	61.90	61.90	61.90
	0.3	63.81	62.86	64.76	62.86	63.81	62.86	61.90	61.90	61.90	61.90	61.90
	0.5	55.24	63.81	65.71	55.24	64.76	62.86	61.90	61.90	61.90	61.90	61.90
Monks3	0	96.07	96.99	97.22	97.22	**97.45**	96.07	96.07	96.76	97.68	96.99	97.22
(432 × 6)	0.05	96.30	96.99	96.76	97.22	**97.45**	96.07	96.53	96.53	96.76	96.99	97.22
	0.1	96.30	96.99	96.76	97.22	**97.45**	96.07	96.53	96.53	96.53	97.22	**97.45**
	0.2	95.83	96.99	96.99	97.22	**97.45**	96.07	96.76	96.76	96.76	96.76	96.76
	0.3	95.60	96.76	96.99	97.22	**97.45**	96.07	96.53	96.53	96.53	96.53	96.53
	0.5	95.60	97.22	97.22	97.22	97.22	96.07	96.07	96.07	96.07	96.07	96.07
Liver Disorder	0	**75.72**	75.14	73.99	74.57	75.14	74.57	75.14	74.57	**75.72**	73.41	72.83
(345 × 6)	0.05	**75.72**	75.14	73.99	74.57	74.57	74.57	74.57	75.14	74.57	74.57	73.41
	0.1	**75.72**	75.14	75.14	75.14	74.57	74.57	**75.72**	75.14	75.14	74.57	73.41
	0.2	**75.72**	75.14	76.30	73.41	**75.72**	74.57	**75.72**	**75.72**	**75.72**	75.14	**75.72**
	0.3	**75.72**	**75.72**	75.14	75.14	74.57	74.57	**75.72**	**75.72**	**75.72**	75.14	75.14
	0.5	75.14	75.14	75.14	75.14	75.14	74.57	75.14	75.14	75.14	75.14	75.14
Planning Relax	0	72.53	71.43	72.53	72.53	73.63	72.53	72.53	72.53	72.53	72.53	72.53
(182 × 12)	0.05	72.53	72.53	72.53	72.53	72.53	72.53	71.43	71.43	72.53	71.43	71.43
	0.1	72.53	72.53	72.53	72.53	72.53	72.53	71.43	71.43	71.43	71.43	71.43
	0.2	72.53	**74.72**	72.53	71.43	71.43	72.53	71.43	71.43	71.43	71.43	71.43
	0.3	71.43	72.53	72.53	72.53	72.53	72.53	72.53	72.53	72.53	72.53	72.53
	0.5	73.63	71.43	**74.72**	71.43	71.43	72.53	71.43	71.43	71.43	71.43	71.43
Fertility	0	88.00	88.00	88.00	88.00	88.00	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**
(100 × 9)	0.05	88.00	88.00	88.00	88.00	88.00	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**
	0.1	88.00	88.00	88.00	88.00	88.00	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**
	0.2	88.00	88.00	88.00	88.00	88.00	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**
	0.3	88.00	88.00	88.00	88.00	88.00	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**	**90.00**
	0.5	88.00	88.00	88.00	88.00	88.00	**90.00**	88.00	88.00	88.00	88.00	88.00

Table 6.3: Accuracy obtained on noise corrupted UCI datasets for non-linear kernel for SPTWSVM

Datasets	r	**Sparse Pin SVM** τ					**TWSVM**	**SPTWSVM** τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Heart-Statlog	0	82.22 ± 0.00	**84.44 ± 0.00**	82.96 ± 0.00	82.96 ± 0.00	80.00 ± 0.00	**84.44 ± 0.00**	**84.44 ± 0.00**	**84.44 ± 0.00**	**84.44 ± 0.00**	**84.44 ± 0.00**	**84.44 ± 0.00**
(270 x 13)	0.05	82.96 ± 1.74	82.81 ± 1.22	83.26 ± 1.44	81.33 ± 0.97	81.19 ± 1.86	82.96 ± 1.17	83.26 ± 0.99	83.26 ± 0.99	83.11 ± 0.97	**83.85 ± 0.97**	83.70 ± 0.74
	0.1	82.07 ± 1.52	81.33 ± 1.62	82.22 ± 0.74	82.67 ± 1.35	81.93 ± 1.86	82.96 ± 1.80	82.96 ± 0.74	82.96 ± 1.54	83.41 ± 1.54	**83.85 ± 0.33**	82.81 ± 0.62
Sonar	0	**63.81 ± 0.00**	56.19 ± 0.00	56.19 ± 0.00	60.95 ± 0.00	60.00 ± 0.00	62.86 ± 0.00	62.86 ± 0.00	61.90 ± 0.00	61.90 ± 0.00	61.90 ± 0.00	61.90 ± 0.00
(208 x 60)	0.05	58.67 ± 6.01	58.67 ± 4.79	57.71 ± 6.01	57.71 ± 4.65	57.52 ± 4.34	58.67 ± 2.85	59.05 ± 0.95	**59.43 ± 0.85**	**59.43 ± 1.59**	**59.43 ± 1.59**	**59.43 ± 1.59**
	0.1	58.48 ± 4.69	59.81 ± 4.60	58.67 ± 3.22	59.24 ± 3.38	55.24 ± 1.20	59.81 ± 1.84	**60.00 ± 1.70**	**60.00 ± 1.70**	59.62 ± 1.66	**60.00 ± 1.70**	59.43 ± 1.77
Monks3	0	96.30 ± 0.00	96.99 ± 0.00	96.76 ± 0.00	97.22 ± 0.00	**97.46 ± 0.00**	96.07 ± 0.00	96.53 ± 0.00	96.53 ± 0.00	96.76 ± 0.00	96.99 ± 0.00	97.22 ± 0.00
(432 x 6)	0.05	85.97 ± 1.74	86.48 ± 0.91	86.48 ± 1.45	85.56 ± 1.95	86.57 ± 2.27	86.53 ± 1.05	86.62 ± 0.84	86.67 ± 0.93	86.67 ± 0.93	86.53 ± 0.90	**86.85 ± 1.02**
	0.1	80.42 ± 2.22	80.46 ± 2.28	80.23 ± 1.89	79.81 ± 1.68	79.90 ± 1.68	80.32 ± 2.86	80.55 ± 2.39	80.51 ± 2.49	80.60 ± 2.13	80.79 ± 2.14	**80.83 ± 1.64**
Liver Disorder	0	**75.72 ± 0.00**	75.14 ± 0.00	73.99 ± 0.00	74.57 ± 0.00	74.57 ± 0.00	74.57 ± 0.00	74.57 ± 0.00	75.14 ± 0.00	74.57 ± 0.00	74.57 ± 0.00	73.41 ± 0.00
(345 x 6)	0.05	73.87 ± 0.48	73.99 ± 0.71	73.99 ± 0.00	73.87 ± 0.48	73.87 ± 0.75	73.41 ± 1.71	73.06 ± 0.66	73.41 ± 0.71	73.64 ± 1.05	**74.16 ± 01.35**	73.06 ± 1.05
	0.1	73.06 ± 1.56	73.41 ± 1.58	73.41 ± 0.71	72.72 ± 1.66	72.72 ± 1.44	72.37 ± 2.50	72.95 ± 1.25	72.95 ± 1.32	**73.53 ± 1.25**	73.29 ± 0.86	71.91 ± 1.86
Planning Relax	0	**72.53 ± 0.00**	72.53 ± 0.00	**72.53 ± 0.00**	72.53 ± 0.00	**72.53 ± 0.00**	**72.53 ± 0.00**	71.43 ± 0.00	71.43 ± 0.00	**72.53 ± 0.00**	71.43 ± 0.00	71.43 ± 0.00
(182 x 12)	0.05	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**	**71.43 ± 0.00**
	0.1	71.43 ± 0.00	71.65 ± 0.49	71.43 ± 0.00	71.65 ± 0.49	**71.87 ± 0.60**	71.65 ± 0.49	71.65 ± 0.49	**71.87 ± 0.60**	71.65 ± 0.49	71.65 ± 0.49	71.65 ± 0.49
Fertility	0	88.00 ± 0.00	88.00 ± 0.00	88.00 ± 0.00	88.00 ± 0.00	88.00 ± 0.00	**90.00 ± 0.00**	**90.00 ± 0.00**	**90.00 ± 0.00**	**90.00 ± 0.00**	**90.00 ± 0.00**	**90.00 ± 0.00**
(100 x 9)	0.05	88.00 ± 0.00	88.00 ± 0.00	88.40 ± 0.89	88.40 ± 0.89	88.00 ± 0.00	88.40 ± 0.89	**88.80 ± 1.10**	88.40 ± 0.89	**88.80 ± 1.10**	**88.80 ± 1.10**	**88.80 ± 1.10**
	0.1	88.40 ± 0.80	88.80 ± 0.98	88.40 ± 0.80	88.40 ± 0.80	88.40 ± 0.80	88.80 ± 0.98	**89.60 ± 0.80**	89.20 ± 0.98	89.20 ± 0.98	89.20 ± 0.98	89.20 ± 0.98



(a) Spect (b) Monk3

(c) Heart-Statlog (d) Liver Disorder

Figure 6.2: 3D surface plots of accuracy of SPTWSVM in relation to ϵ and τ. Subfigures (a) and (b) correspond to the linear case whereas subfigures (c) and (d) correspond to the non-linear case.

A similar analysis has been done for the non-linear case with an RBF kernel,

$$K(x^{(i)}, x^{(j)}) = exp\left(-\gamma||x^{(i)} - x^{(j)}||^2\right), \gamma > 0,$$

and the corresponding results are reported in Table 6.2. Here, we choose the optimal values of $\gamma \in \{10^i : i = -7, -6, -5, ..., +1, +2, +3\}$ and c according to ten-fold cross validation as earlier. Similar to the earlier table, we highlight the best result in view of accuracy in bold. From Table 6.1 and Table 6.2 we learn that the novel SPTWSVM yields the best predicition accuracy for eight datasets. Hence, in general, the accuracy obtained by SPTWSVM matches and outperforms those of other models.

Now, we corrupt the features of six benchmark UCI datasets with zero-mean Gaussian noise. Both the training and testing datasets are perturbed by the same noise. For each feature, the ratio of variance of noise to that of feature is denoted by r. We use an RBF kernel and use Sparse Pin SVM, TWSVM and SPTWSVM to perform classification on the corrupted datasets for different levels of noise. For each combination of r and τ, the experiments are repeated five times and the average and standard deviation of the accuracies obtained have been reported in Table 6.3. Here, ϵ is kept constant, equal to 0.05, for the sake of easier data representation. As earlier, for each dataset and different r value, the model with the best average classification accuracy is highlighted in bold. One can observe from Table 6.3 that our SPTWSVM achieves better results, that is, the average accuracy of SPTWSVM is the highest or at par with the other compared models for fifteen out of eighteen cases. In addition, the standard deviation is small which indicates that SPTWSVM is noise insensitive, which supports our theoretical analysis.

In Table 6.4 and Table 6.5, the sparsity of our proposed SPTWSVM is analyzed as compared

Table 6.4: Sparsity on UCI datasets with linear kernel for SPTWSVM

Datasets	ϵ	**TWSVM**		**SPTWSVM**	
		τ = 0		τ = 0.5	
Heart-Statlog	0	60	75	60	75
(270 × 13)	0.05			49	57
	0.1			37	36
	0.2			29	26
	0.3			24	20
	0.5			16	19
Australian	0	148	113	191	153
(690 × 14)	0.05			145	148
	0.1			138	134
	0.2			130	122
	0.3			123	119
	0.5			100	100
Heart-C	0	27	26	82	69
(303 × 13)	0.05			59	54
	0.1			41	39
	0.2			24	26
	0.3			23	20
	0.5			18	16
SPECT	0	40	30	40	40
(267 × 22)	0.05			22	24
	0.1			18	24
	0.2			17	25
	0.3			14	22
	0.5			16	17
Monk3	0	30	26	62	60
(432 × 6)	0.05			54	49
	0.1			42	42
	0.2			34	25
	0.3			24	23
	0.5			22	20
Breast	0	23	26	32	26
(116 ×10)	0.05			29	26
	0.1			28	26
	0.2			26	26
	0.3			23	26
	0.5			21	25

Table 6.5: Sparsity on UCI datasets with non-linear kernel for SPTWSVM

Datasets	ϵ	**TWSVM**		**SPTWSVM**	
		τ=0		τ=0.5	
Heart-Statlog	0	60	74	60	75
(270 × 13)	0.05			48	51
	0.1			40	34
	0.2			27	27
	0.3			21	22
	0.5			20	20
Sonar	0	29	42	55	48
(208 × 60)	0.05			5	10
	0.1			5	10
	0.2			5	10
	0.3			7	10
	0.5			5	13
Monk3	0	62	60	62	60
(432 × 6)	0.05			13	11
	0.1			11	9
	0.2			8	8
	0.3			9	8
	0.5			7	23
Liver Disorder	0	59	52	100	72
(345 × 6)	0.05			79	61
	0.1			67	55
	0.2			55	48
	0.3			49	45
	0.5			42	40
Planning Relax	0	26	65	26	65
(182 × 12)	0.05			26	63
	0.1			26	60
	0.2			26	54
	0.3			26	49
	0.5			26	47
Fertility	0	6	41	6	44
(100 × 9)	0.05			6	42
	0.1			6	41
	0.2			6	41
	0.3			6	40
	0.5			6	38

to the original TWSVM for the linear and non-linear cases, respectively. In both tables, the two columns under each model show the number of non-zero dual variables corresponding to each of the two separating hyperplanes. Here, we have kept c and γ (for RBF kernel) constant for all ϵ values which resists the effect of change in hyperparameters. Noticing the results, in general, as ϵ increases we can observe that the sparsity of our solution increases, which is expected since the sub-gradients of a lot of the error terms in our dual formulation become zero. From both the tables it is evident that our novel SPTWSVM is more sparse as compared to the original TWSVM while simultaneously maintaining noise-insensitive properties. This sparsity of solution makes the the prediction process faster than the TWSVM which is of immense value, especially in datasets with large samples.

We also plot the effect of hyperparameter selection on accuracies obtained for four different datasets in Fig. 6.2. The figure shows the 3D surface plots of accuracy in relation to ϵ and τ. In Fig. 6.2(a), a spike can be seen which shows a drastic increase in accuracy from 78% to approximately 91% with a slight variation in the value of ϵ and τ. This drastic change in the value of accuracy, demonstrates the sensitivity of the model with respect to parameter selection. In a similar fashion, we can visualize the sensitivity of model performance based on optimal parameters in subfigures (b), (c) and (d). Therefore, parameter selection becomes an important issue while calculating the performance of our SPTWSVM model.

Table 6.6 summarizes the results for a linear kernel, $K(x, y) = \phi(x)^T\phi(y)$, on six different UCI datasets. Here, we compare the accuracy of our second ISPTWSVM model with that of the Sparse Pin SVM and the TWSVM. The optimal value of c and c' for each (ϵ, τ) combination for ISPTWSVM is computed using ten-fold cross-validation and same is the case with Sparse Pin SVM and TWSVM. For each dataset, we apply Sparse Pin SVM, TWSVM and ISPTWSVM to perform classification for different (ϵ, τ) combinations, and subsequently the model with the best classification accuracy is highlighted in bold. The table's representation style is similar to that of Table 6.1. From the results tabulated in Table 6.6, we see that classification performance of ISPTWSVM is better than that of Sparse Pin SVM and TWSVM in most of the datasets which is expected since ISPTWSVM retains all previous enhancements of SPTWSVM. Similar analysis is done for the non-linear case using RBF Kernel and corresponding results are tabulated in Table 6.7, with the maximum accuracy being highlighted in bold. The results show that ISPTWSVM possesses all the positive aspects of SPTWSVM while being feasible for large scale datasets.

To highlight optimal parameters for the performed experiments, we provide the optimal values of c and γ for Table 6.1 and Table 6.2, corresponding to the SPTWSVM model, in Table 6.8, Table 6.9 and Table 6.10. Similarly, the optimal values of c, c' and γ for Table 6.6 and Table 6.7, corresponding to the ISPTWSVM model, are presented in Table 6.11, Table 6.12 and Table 6.13.

Table 6.6: ISPTWSVM performance on UCI datasets for linear case

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**ISPTWSVM**				
		τ						τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Sonar	0.00	54.29	54.29	54.29	54.29	54.29	54.29	60.95	60.00	57.14	59.05	59.05
(208 x 60)	0.05	54.29	54.29	54.29	54.29	54.29	54.29	61.90	61.90	60.95	59.05	58.10
	0.10	54.29	54.29	54.29	54.29	54.29	54.29	**63.81**	60.95	60.95	59.05	58.10
	0.20	54.29	54.29	54.29	54.29	54.29	54.29	62.86	60.00	60.00	59.05	59.05
	0.30	54.29	54.29	54.29	54.29	54.29	54.29	60.95	60.00	60.00	60.95	59.05
	0.50	54.29	54.29	54.29	54.29	54.29	54.29	60.00	61.90	60.00	60.00	60.00
Fertility	0.00	88.00	88.00	88.00	88.00	88.00	88.00	**92.00**	88.00	88.00	88.00	90.00
(100 x 9)	0.05	88.00	88.00	88.00	88.00	88.00	88.00	90.00	88.00	90.00	88.00	**92.00**
	0.10	88.00	88.00	88.00	88.00	88.00	88.00	88.00	88.00	88.00	88.00	90.00
	0.20	88.00	88.00	88.00	88.00	88.00	88.00	90.00	90.00	90.00	90.00	90.00
	0.30	88.00	88.00	88.00	88.00	88.00	88.00	90.00	90.00	90.00	90.00	90.00
	0.50	88.00	88.00	88.00	88.00	88.00	88.00	90.00	90.00	90.00	88.00	90.00
SPECT	0.00	73.80	74.33	75.40	72.73	72.73	78.61	91.44	93.58	90.91	93.05	86.10
(267 x 22)	0.05	74.33	74.33	74.33	72.73	72.73	78.61	90.91	93.58	90.91	93.05	86.10
	0.10	74.33	73.26	74.33	72.73	72.73	78.61	90.91	**94.12**	90.91	93.05	86.10
	0.20	74.33	74.33	74.87	72.73	72.73	78.61	89.84	93.58	90.91	93.05	86.10
	0.30	74.33	73.80	74.87	73.26	72.73	78.61	91.98	93.05	90.91	93.05	86.63
	0.50	73.80	73.80	74.87	75.40	74.87	78.61	90.37	93.58	90.91	92.51	86.63
Monks2	0.00	67.13	67.13	67.13	67.13	67.13	66.43	**71.76**	67.13	67.36	67.13	67.36
(432 x 6)	0.05	67.13	67.13	67.13	67.13	67.13	66.43	67.13	67.13	67.36	67.59	67.36
	0.10	67.13	67.13	67.13	67.13	67.13	66.43	68.06	67.13	67.36	67.13	67.13
	0.20	67.13	67.13	67.13	67.13	67.13	66.43	68.29	67.13	67.36	67.13	67.13
	0.30	67.13	67.13	67.13	67.13	67.13	66.43	67.13	67.13	67.36	67.13	67.13
	0.50	67.13	67.13	67.13	67.13	67.13	66.43	67.13	67.36	67.13	67.13	67.13
Haberman	0.00	75.97	73.38	73.38	73.38	73.38	69.48	75.97	73.38	73.38	73.38	73.38
(306 x 3)	0.05	75.97	75.32	73.38	73.38	73.38	69.48	**76.62**	73.38	75.32	73.38	73.38
	0.10	**76.62**	75.97	73.38	73.38	73.38	69.48	73.38	73.38	73.38	73.38	73.38
	0.20	75.97	75.97	75.97	75.32	74.03	69.48	74.03	73.38	74.68	75.32	75.32
	0.30	75.97	75.97	75.97	75.97	75.97	69.48	73.38	73.38	74.68	73.38	75.97
	0.50	73.38	73.38	73.38	73.38	73.38	69.48	74.68	73.38	75.97	74.03	73.38
Planning Relax	0.00	71.43	71.43	71.43	71.43	71.43	68.13	71.43	71.43	69.23	64.83	68.13
(182 x 12)	0.05	71.43	71.43	71.43	71.43	71.43	69.48	71.43	72.53	69.23	64.83	68.13
	0.10	71.43	71.43	71.43	71.43	71.43	69.48	71.43	71.43	69.23	64.83	68.13
	0.20	71.43	71.43	71.43	71.43	71.43	69.48	71.43	71.43	67.03	64.83	68.13
	0.30	71.43	71.43	71.43	71.43	71.43	69.48	**74.72**	71.43	67.03	64.83	68.13
	0.50	71.43	71.43	71.43	71.43	71.43	69.48	72.53	71.43	67.03	64.83	68.13

Table 6.7: ISPTWSVM performance on UCI datasets for non-linear case

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**ISPTWSVM**				
		τ						τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Heart-Statlog	0.00	83.70	84.44	82.22	82.96	80.74	84.44	**84.56**	**84.56**	**84.56**	**84.56**	**84.56**
(270 x 13)	0.05	82.22	84.44	82.96	82.96	80.00	84.44	82.56	82.56	**84.56**	**84.56**	**84.56**
	0.10	81.48	84.44	82.96	82.96	81.48	84.44	**84.56**	**84.56**	**84.56**	**84.56**	**84.56**
	0.20	82.22	85.18	82.96	82.96	83.70	84.44	82.56	**84.56**	**84.56**	**84.56**	**84.56**
	0.30	82.22	84.44	82.96	82.22	84.44	84.44	**84.56**	82.56	**84.56**	**84.56**	**84.56**
	0.50	83.70	84.44	85.18	83.70	83.70	84.44	82.56	**84.56**	**84.56**	**84.56**	**84.56**
Heart-C	0.00	79.61	80.26	77.63	78.95	77.63	**82.89**	81.95	81.95	81.95	81.95	81.95
(303x13)	0.05	80.92	80.92	78.29	78.95	78.95	**82.89**	81.95	81.95	81.95	81.95	81.95
	0.10	80.92	80.26	78.29	78.29	78.95	**82.89**	81.95	81.95	81.95	81.95	81.95
	0.20	80.26	79.61	78.95	77.63	78.95	**82.89**	81.95	81.95	81.95	81.95	81.95
	0.30	80.26	79.61	78.95	78.29	80.26	**82.89**	81.95	81.95	81.95	81.95	81.95
	0.50	82.24	80.26	80.26	80.26	80.26	**82.89**	81.95	81.95	81.95	81.95	81.95
SPECT	0.00	91.98	91.98	91.98	91.98	91.98	91.44	**92.40**	91.98	91.98	91.98	91.98
(267 x 22)	0.05	91.98	91.98	91.98	91.98	91.98	91.44	91.98	91.98	91.98	91.98	91.98
	0.10	91.98	91.98	91.98	91.98	91.98	91.44	91.98	91.98	91.98	91.98	91.98
	0.20	91.98	91.98	91.98	91.98	91.98	91.44	91.98	91.98	91.98	91.98	91.98
	0.30	91.98	91.98	91.98	91.98	91.98	91.44	91.98	91.98	91.98	91.98	91.98
	0.50	91.98	91.98	91.98	91.98	91.98	91.44	91.98	91.98	91.98	91.98	91.98
Haberman	0.00	76.62	77.27	**77.92**	77.27	77.27	**77.92**	**77.92**	73.38	73.38	**77.92**	73.38
(306 x 3)	0.05	77.27	76.62	76.62	76.62	76.62	**77.92**	**77.92**	73.38	73.38	**77.92**	73.38
	0.10	76.62	77.27	75.97	76.62	77.27	**77.92**	73.38	75.38	73.38	73.38	73.38
	0.20	75.97	76.62	76.62	76.62	75.97	**77.92**	73.38	73.38	75.38	73.38	73.38
	0.30	77.27	77.27	77.27	77.27	77.27	**77.92**	73.38	**77.92**	73.38	**77.92**	73.38
	0.50	76.62	76.62	76.62	76.62	76.62	**77.92**	**77.92**	73.38	73.38	73.38	75.38
Planning Relax	0.00	72.53	71.43	72.53	72.53	73.63	72.53	71.43	72.61	72.61	71.43	71.43
(182 x 12)	0.05	72.53	72.53	72.53	72.53	72.53	72.53	71.43	71.43	**72.61**	71.43	72.61
	0.10	72.53	72.53	72.53	72.53	72.53	72.53	71.43	71.43	71.43	71.43	71.43
	0.20	72.53	74.72	72.53	71.43	71.43	72.53	71.43	71.43	72.61	71.43	71.43
	0.30	71.43	72.53	72.53	72.53	72.53	72.53	71.43	71.43	72.61	71.43	71.43
	0.50	73.63	71.43	74.72	71.43	71.43	72.53	71.43	71.43	72.61	71.43	71.43
Fertility	0.00	88.00	88.00	88.00	88.00	88.00	**90.00**	88.00	**90.00**	**90.00**	88.00	**90.00**
(100 x 9)	0.05	88.00	88.00	88.00	88.00	88.00	**90.00**	**90.00**	**90.00**	**90.00**	88.00	88.00
	0.10	88.00	88.00	88.00	88.00	88.00	**90.00**	88.00	**90.00**	**90.00**	88.00	88.00
	0.20	88.00	88.00	88.00	88.00	88.00	**90.00**	88.00	**90.00**	**90.00**	88.00	**90.00**
	0.30	88.00	88.00	88.00	88.00	88.00	**90.00**	88.00	**90.00**	**90.00**	88.00	88.00
	0.50	88.00	88.00	88.00	88.00	88.00	**90.00**	88.00	**90.00**	**90.00**	88.00	88.00

Table 6.8: Optimal c values for linear kernel for SPTWSVM

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**SPTWSVM**				
		τ						τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Heart-Statlog	0	1	1	10^{3}	10^{-1}	10^{-2}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
(270 × 13)	0.05	1	1	10^{1}	10^{1}	10^{-2}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.1	10^{-1}	10^{4}	10^{1}	1	10^{-2}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	1	10^{-5}
	0.20	10^{-1}	10^{2}	10^{5}	10^{-1}	10^{3}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.30	10^{-1}	10^{-1}	1	10^{1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.50	10^{-1}	10^{-1}	10^{-1}	10^{-1}	1	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
Australian	0	10^{4}	10^{-1}	10^{3}	10^{-1}	10^{-2}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
(690 × 14)	0.05	10^{1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}
	0.10	10^{3}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
	0.20	10^{5}	10^{-1}	10^{-1}	10^{-1}	10^{-2}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
	0.30	10^{5}	10^{-1}	10^{-1}	10^{-1}	10^{5}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
	0.50	10^{5}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	1	1	1	1	1
Heart-C	0	1	10^{3}	1	10^{-3}	10^{-3}	1	1	10^{-5}	1	10^{-5}	10^{-5}
(303 × 13)	0.05	1	1	10^{3}	10^{-4}	10^{-4}	1	1	1	1	10^{-5}	10^{-5}
	0.10	1	1	1	10^{-3}	10^{-4}	1	1	1	1	10^{-5}	10^{-5}
	0.20	10^{-1}	10^{1}	10^{-1}	10^{-4}	10^{-4}	1	1	1	1	1	10^{-5}
	0.30	10^{-1}	1	1	1	10^{-3}	1	1	1	1	1	1
	0.50	10^{-1}	10^{-1}	10^{3}	10^{5}	10^{1}	1	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
SPECT	0	10^{-1}	10^{2}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-3}	10^{-3}	10^{-4}	10^{-5}
(267 × 22)	0.05	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{3}	10^{-4}	10^{-4}	10^{-5}
	0.1	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-4}	10^{-4}	10^{-5}
	0.2	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-4}	10^{-4}	10^{-5}
	0.3	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-5}
	0.50	10^{-2}	10^{-2}	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
Monk3	0	10^{-1}	10^{1}	1	10^{4}	10^{-1}	1	1	1	1	1	10^{-5}
(432 × 6)	0.05	10^{-1}	1	1	10^{1}	10^{-1}	1	1	1	1	1	1
	0.1	10^{-1}	10^{1}	10^{5}	10^{-1}	10^{-1}	1	1	1	1	1	1
	0.2	10^{-2}	10^{-1}	10^{1}	10^{1}	1	1	1	1	1	1	1
	0.3	1	10^{-1}	10^{-1}	10^{5}	10^{4}	1	1	1	1	10^{-4}	1
	0.5	10^{-1}	1	1	10^{-1}	10^{-1}	1	1	1	1	1	1
Breast	0	10^{-3}	10^{-3}	1	10^{-3}	10^{-3}	10^{-1}	10^{1}	10^{-1}	1	1	10^{-1}
(116 × 10)	0.05	10^{-3}	10^{-3}	10^{-1}	10^{-3}	10^{-5}	10^{-1}	10^{1}	10^{1}	10^{2}	1	1
	0.1	10^{-3}	10^{-3}	1	10^{-3}	10^{-3}	10^{-1}	10^{-1}	10^{1}	10^{-1}	10^{-1}	10^{-1}
	0.2	10^{-3}	10^{-3}	10^{2}	10^{-3}	1	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}
	0.3	10^{-3}	10^{-3}	10^{-2}	10^{2}	1	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}
	0.5	10^{-3}	10^{-3}	10^{-2}	10^{-1}	10^{-3}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{1}

Table 6.9: Optimal c values for non-linear kernel for SPTWSVM

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**SPTWSVM**				
		τ						τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Heart-Statlog	0	10^4	10^5	10^3	10^5	10^4	10^{-2}	1	1	1	1	1
(270 × 13)	0.05	10^5	10^5	10^5	10^5	10^4	10^{-2}	10^{-2}	10^{-2}	1	1	10^{-1}
	0.1	10^4	10^5	10^5	10^5	10^5	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}
	0.20	10^4	10^5	10^3	10^4	10^5	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-1}
	0.30	10^4	10^5	10^4	10^4	10^4	10^{-2}	1	1	1	1	1
	0.50	10^5	10^4	10^4	10^4	10^4	10^{-2}	1	1	1	1	1
Sonar	0	10^3	1	10	1	1	10^{-3}	10^3	10^2	10^2	10^2	10^2
(208 × 60)	0.05	10^1	1	1	10^2	10^4	10^{-3}	10^{-3}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.10	10^4	10^4	10^1	10^3	10^2	10^{-3}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.20	10^{-2}	10^3	10^3	10^1	10^2	10^{-3}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.30	10^4	10^3	10^4	10^4	10^2	10^{-3}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.50	10^{-2}	10^3	10^3	10^{-2}	10^4	10^{-3}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
Monk3	0	10^1	10^1	10^4	10^1	10^1	10^{-5}	10^{-5}	1	10^1	1	1
(432 × 6)	0.05	10^1	10^1	10^1	10^2	10^1	10^{-5}	10^{-2}	10^{-2}	1	1	1
	0.10	10^1	10^1	10^1	10^2	10^1	10^{-5}	10^{-2}	10^{-2}	10^{-2}	1	1
	0.20	10^1	10^1	10^1	10^1	10^1	10^{-5}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}
	0.30	10^1	10^1	10^1	10^1	10^1	10^{-5}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}
	0.50	1	10^4	10^2	10^2	10^2	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
Liver Disorder	0	1	1	1	10^2	10^4	1	1	1	1	1	10^{-1}
(345 × 6)	0.05	1	1	1	1	10^2	1	1	1	1	1	1
	0.1	1	1	1	1	10^4	1	1	1	1	1	1
	0.2	1	1	10^2	10^5	10^4	1	1	1	1	1	1
	0.3	1	1	1	1	10^4	1	1	1	1	1	1
	0.50	1	1	1	1	1	1	1	1	1	1	1
Planning Relax	0	1	10^{-5}	1	1	10^5	1	1	1	1	1	1
(182 × 12)	0.05	1	1	1	1	1	1	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.1	1	1	1	1	1	1	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.2	1	10^5	1	10^{-5}	10^{-5}	1	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.3	10^{-5}	1	10^3	1	1	1	1	1	1	1	1
	0.5	10^5	1	10^5	10^{-5}	10^{-5}	1	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
Fertility	0	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^5	10^{-2}	10^{-2}	10^1	10^1	10^2	10^1
(100 × 9)	0.05	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^2	10^1
	0.1	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-2}	10^1	10^{-2}	10^{-2}	10^{-2}	10^{-2}
	0.2	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}
	0.3	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^1	10^{-2}
	0.5	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-2}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}

Table 6.10: Optimal γ values for non-linear kernel for SPTWSVM

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**SPTWSVM**				
		τ						τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Heart-Statlog	0	10^{-6}	10^{-6}	10^{-5}	10^{-6}	10^{-5}	10^{-5}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
(270 × 13)	0.05	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-6}	10^{-6}	10^{-5}
	0.1	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.20	10^{-6}	10^{-6}	10^{-5}	10^{-6}	10^{-6}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.30	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-5}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
	0.50	10^{-7}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-5}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
Sonar	0	10^{-6}	1	10^{-4}	1	1	10^{-3}	10^{-7}	10^{-7}	10^{-6}	10^{-6}	10^{-6}
(208 × 60)	0.05	10^{-4}	1	1	10^{-5}	10^{-7}	10^{-3}	10^{-3}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.10	10^{-7}	10^{-7}	10^{-4}	10^{-6}	10^{-5}	10^{-3}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.20	10^{-1}	10^{-6}	10^{-6}	10^{-4}	10^{-5}	10^{-3}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.30	10^{-7}	10^{-6}	10^{-7}	10^{-7}	10^{-5}	10^{-3}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.50	10^{-1}	10^{-6}	10^{-6}	10^{-1}	10^{-7}	10^{-3}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
Monk 3	0	10^{-1}	10^{-1}	10^{-3}	10^{-1}	10^{-1}	10^{-3}	10^{-3}	10^{-3}	10^{-4}	10^{-2}	10^{-2}
(432 × 6)	0.05	10^{-1}	10^{-1}	10^{-2}	10^{-2}	10^{-1}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}
	0.10	10^{-1}	10^{-1}	10^{-1}	10^{-2}	10^{-1}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}
	0.20	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}
	0.30	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}
	0.50	10^{-1}	10^{-3}	10^{-2}	10^{-2}	10^{-2}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}
Liver Disorder	0	10^{-4}	10^{-4}	10^{-4}	10^{-5}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-5}
(345 × 6)	0.05	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-5}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-5}
	0.1	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
	0.2	10^{-4}	10^{-4}	10^{-5}	10^{-7}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
	0.3	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
	0.5	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
Planning Relax	0	10^{-1}	10^{-7}	10^{-1}	1	10^{-3}	10^{-4}	10^{-4}	1	10^{-4}	10^{-4}	10^{-4}
(182 × 12)	0.05	1	1	1	1	1	10^{-4}	10^{-7}	10^{-7}	10^{-1}	10^{-7}	10^{-7}
	0.1	1	1	1	1	1	10^{-4}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}
	0.2	1	10^{-3}	10^{1}	10^{-7}	10^{-7}	10^{-4}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}
	0.3	10^{-7}	1	10^{-2}	1	1	10^{-4}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.5	10^{-3}	10^{-7}	10^{-3}	10^{-7}	10^{-7}	10^{-4}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}
Fertility	0	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-4}	10^{-4}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
(100 × 9)	0.05	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-7}	10^{-7}
	0.1	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.2	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.3	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.5	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}

Table 6.11: Optimal c and c' values corresponding to linear case of ISPTWSVM

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**ISPTWSVM ($c_1 = c_3 = c$)**					**ISPTWSVM ($c_2 = c_4 = c'$)**				
		τ						τ					τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1	0.01	0.1	0.2	0.5	1
Sonar	0.00	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{3}	10^{1}	10^{0}	10^{0}	10^{0}	10^{-5}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
(208 x 60)	0.05	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}		10^{0}	10^{1}	10^{4}	10^{4}	10^{1}	10^{0}	10^{-1}	10^{-2}	10^{-2}	10^{-2}
	0.10	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}		10^{4}	10^{1}	10^{3}	10^{5}	10^{3}	10^{0}	10^{-1}	10^{-2}	10^{-2}	10^{-2}
	0.20	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}		10^{0}	10^{3}	10^{1}	10^{1}	10^{5}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-4}
	0.30	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}		10^{0}	10^{3}	10^{0}	10^{0}	10^{1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.50	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}		10^{0}	10^{0}	10^{0}	10^{0}	10^{3}	10^{0}	10^{0}	10^{0}	10^{0}	10^{3}
Fertility	0.00	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
(100 x 9)	0.05	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.10	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.20	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{1}	10^{0}	10^{0}	10^{0}
	0.30	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{1}	10^{3}	10^{1}	10^{0}	10^{1}
	0.50	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{1}	10^{1}	10^{1}	10^{0}	10^{1}
SPECT	0.00	10^{-1}	10^{2}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{5}	10^{1}	10^{1}	10^{2}	10^{2}
(267 x 22)	0.05	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{3}	10^{2}	10^{4}	10^{2}	10^{2}
	0.10	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{-5}	10^{0}	10^{1}	10^{0}	10^{0}	10^{0}
	0.20	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{1}	10^{3}	10^{1}	10^{0}	10^{1}
	0.30	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{1}	10^{1}	10^{1}	10^{0}	10^{1}
	0.50	10^{-2}	10^{-2}	10^{-1}	10^{-1}	10^{-1}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{3}	10^{2}	10^{4}	10^{2}	10^{2}
Monk2	0.00	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{4}	10^{1}	10^{1}	10^{4}	10^{1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
	0.05	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{0}	10^{0}	10^{0}	10^{0}	10^{0}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.10	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-1}	10^{-1}	10^{4}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.20	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-1}	10^{-1}	10^{4}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.30	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-1}	10^{3}	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{3}	10^{-1}	10^{-1}	10^{-1}
	0.50	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{5}	10^{-5}	10^{3}	10^{5}	10^{1}	10^{1}	10^{2}	10^{2}
Haberman	0.00	10^{-2}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-1}	10^{4}	10^{1}	10^{1}	10^{4}	10^{1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
(306 x 3)	0.05	10^{-2}	10^{-1}	10^{-5}	10^{-5}	10^{-5}		10^{0}	10^{0}	10^{0}	10^{0}	10^{0}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.10	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}		10^{-1}	10^{-1}	10^{4}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.20	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}		10^{-1}	10^{-1}	10^{4}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.30	10^{-1}	10^{-1}	10^{-1}	10^{-1}	10^{-1}		10^{-1}	10^{3}	10^{-1}	10^{-1}	10^{-1}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{2}
	0.50	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{5}	10^{-5}	10^{3}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}
Planning Relax	0.00	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{4}	10^{1}	10^{1}	10^{4}	10^{2}	10^{4}	10^{1}	10^{1}	10^{4}	10^{2}
(182 x 12)	0.05	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{2}	10^{2}	10^{3}	10^{5}
	0.10	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{2}	10^{4}	10^{4}	10^{3}	10^{3}
	0.20	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{4}	10^{2}	10^{5}	10^{3}	10^{5}
	0.30	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{2}	10^{5}	10^{3}	10^{5}	10^{-5}
	0.50	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}

Table 6.12: Optimal c and c' values corresponding to non-linear case of ISPTWSVM

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**ISPTWSVM ($c_1 = c_3 = c$)**					**ISPTWSVM (c_2=$c_4 = c'$)**				
		τ						τ					τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1	0.01	0.1	0.2	0.5	1
Heart-Statlog	0.00	10^{4}	10^{5}	10^{3}	10^{5}	10^{4}	10^{-2}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}	10^{3}	10^{1}	10^{1}	10^{1}	10^{1}
(270 x 13)	0.05	10^{5}	10^{5}	10^{5}	10^{5}	10^{4}		10^{-2}	10^{-2}	10^{0}	10^{0}	10^{-1}	10^{0}	10^{1}	10^{1}	10^{1}	10^{1}
	0.10	10^{4}	10^{5}	10^{5}	10^{5}	10^{5}		10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{0}	10^{4}	10^{5}	10^{0}	10^{0}
	0.20	10^{4}	10^{5}	10^{3}	10^{4}	10^{5}		10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-1}	10^{3}	10^{2}	10^{2}	10^{5}	10^{4}
	0.30	10^{4}	10^{5}	10^{4}	10^{4}	10^{4}		10^{0}	10^{0}	10^{0}	10^{0}	10^{0}	10^{4}	10^{2}	10^{4}	10^{1}	10^{1}
	0.50	10^{5}	10^{4}	10^{4}	10^{4}	10^{4}		10^{0}	10^{0}	10^{0}	10^{0}	10^{0}	10^{2}	10^{4}	10^{1}	10^{2}	10^{5}
Heart-C	0.00	10^{3}	10^{5}	10^{2}	10^{2}	10^{2}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{4}	10^{2}	10^{4}	10^{1}	10^{1}
(303x13)	0.05	10^{4}	10^{5}	10^{4}	10^{2}	10^{2}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{2}	10^{4}	10^{1}	10^{2}	10^{5}
	0.10	10^{4}	10^{5}	10^{5}	10^{4}	10^{4}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{1}	10^{4}	10^{2}	10^{1}	10^{2}
	0.20	10^{4}	10^{2}	10^{2}	10^{2}	10^{4}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}
	0.30	10^{2}	10^{2}	10^{2}	10^{2}	10^{3}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}
	0.50	10^{3}	10^{2}	10^{2}	10^{4}	10^{4}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{-5}	10^{0}	10^{0}	10^{5}
SPECT	0.00	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{1}	10^{4}	10^{3}	10^{3}	10^{3}	10^{2}	10^{4}	10^{1}	10^{1}	10^{4}	10^{2}
(267 x 22)	0.05	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{1}	10^{1}	10^{1}	10^{1}	10^{-5}	10^{2}	10^{2}	10^{3}	10^{5}
	0.10	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{1}	10^{1}	10^{1}	10^{1}	10^{1}	10^{2}	10^{4}	10^{4}	10^{3}	10^{3}
	0.20	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{1}	10^{1}	10^{1}	10^{1}	10^{2}	10^{4}	10^{2}	10^{5}	10^{3}	10^{5}
	0.30	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{1}	10^{1}	10^{1}	10^{1}	10^{3}	10^{2}	10^{5}	10^{3}	10^{5}	10^{-5}
	0.50	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{1}	10^{1}	10^{1}	10^{2}	10^{5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
Haberman	0.00	10^{3}	10^{2}	10^{2}	10^{5}	10^{4}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}
(306 X 3)	0.05	10^{4}	10^{2}	10^{4}	10^{1}	10^{1}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{-4}	10^{-4}	10^{-3}	10^{1}
	0.10	10^{2}	10^{4}	10^{1}	10^{2}	10^{5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{-3}	10^{0}	10^{-5}	10^{-5}
	0.20	10^{1}	10^{4}	10^{2}	10^{1}	10^{2}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}
	0.30	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}
	0.50	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}
Planning Relax	0.00	10^{0}	10^{-5}	10^{0}	10^{0}	10^{5}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}	10^{4}	10^{-1}	10^{3}	10^{-1}	10^{-2}
(182 x 12)	0.05	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{1}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
	0.10	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{3}	10^{-1}	10^{-1}	10^{-1}	10^{-1}
	0.20	10^{0}	10^{5}	10^{0}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{5}	10^{-1}	10^{-1}	10^{-1}	10^{-2}
	0.30	10^{-5}	10^{0}	10^{3}	10^{0}	10^{0}		10^{0}	10^{0}	10^{0}	10^{0}	10^{0}	10^{5}	10^{-1}	10^{-1}	10^{-1}	10^{5}
	0.50	10^{5}	10^{-5}	10^{5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{5}	10^{-1}	10^{-1}	10^{-1}	10^{-1}
Fertility	0.00	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-2}	10^{-2}	10^{1}	10^{1}	10^{1}	10^{1}	10^{-3}	10^{2}	10^{2}	10^{-3}	10^{-3}
(100 x 9)	0.05	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-2}	10^{-2}	10^{-2}	10^{1}	10^{1}	10^{-3}	10^{0}	10^{4}	10^{-3}	10^{-3}
	0.10	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{5}	10^{5}	10^{5}	10^{-3}	10^{-3}
	0.20	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{2}	10^{-1}	10^{0}	10^{-3}	10^{-3}
	0.30	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{0}	10^{5}	10^{-3}	10^{-3}	10^{-3}
	0.50	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-1}	10^{-1}	10^{2}	10^{-2}	10^{0}

Table 6.13: Optimal γ values for non-linear kernel of ISPTWSVM

Datasets	ϵ	**Sparse Pin SVM**					**TWSVM**	**SPTWSVM**				
		τ						τ				
		0.01	0.1	0.2	0.5	1		0.01	0.1	0.2	0.5	1
Heart-Statlog (270 x 13)	0.00	10^{-6}	10^{-6}	10^{-5}	10^{-6}	10^{-5}	10^{-5}	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
	0.05	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-5}		10^{-7}	10^{-7}	10^{-6}	10^{-6}	10^{-6}
	0.10	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}		10^{-3}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.20	10^{-6}	10^{-6}	10^{-5}	10^{-6}	10^{-6}		10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.30	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}		10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
	0.50	10^{-7}	10^{-6}	10^{-6}	10^{-6}	10^{-6}		10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
Heart-C (303x13)	0.00	10^{-5}	10^{-6}	10^{-4}	10^{-4}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.05	10^{-6}	10^{-6}	10^{-5}	10^{-4}	10^{-5}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.10	10^{-6}	10^{-6}	10^{-6}	10^{-5}	10^{-7}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.20	10^{-6}	10^{-4}	10^{-4}	10^{-4}	10^{-7}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.30	10^{-4}	10^{-4}	10^{-4}	10^{-5}	10^{-6}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.50	10^{-5}	10^{-5}	10^{-5}	10^{-7}	10^{-6}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
SPECT (267 x 22)	0.00	10^{1}	10^{1}	10^{1}	10^{1}	10^{1}	10^{2}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}
	0.05	10^{1}	10^{1}	10^{1}	10^{1}	10^{1}		10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}
	0.10	10^{1}	10^{1}	10^{1}	10^{1}	10^{1}		10^{2}	10^{-7}	10^{-7}	10^{-7}	10^{-7}
	0.20	10^{1}	10^{1}	10^{1}	10^{1}	10^{1}		10^{2}	10^{-7}	10^{2}	10^{2}	10^{2}
	0.30	10^{1}	10^{1}	10^{1}	10^{1}	10^{1}		10^{2}	10^{-7}	10^{2}	10^{2}	10^{2}
	0.50	10^{1}	10^{1}	10^{1}	10^{1}	10^{1}		10^{2}	10^{1}	10^{1}	10^{2}	10^{2}
Haberman (306 X 3)	0.00	10^{-4}	10^{-3}	10^{-3}	10^{-4}	10^{-3}	10^{-4}	10^{-4}	10^{-3}	10^{-4}	10^{-3}	10^{-4}
	0.05	10^{-4}	10^{-3}	10^{-4}	10^{-3}	10^{-3}		10^{-4}	10^{-4}	10^{-3}	10^{-3}	10^{-4}
	0.10	10^{-3}	10^{-4}	10^{-3}	10^{-3}	10^{-4}		10^{-4}	10^{-4}	10^{-3}	10^{-3}	10^{-4}
	0.20	10^{-3}	10^{-4}	10^{-3}	10^{-3}	10^{-3}		10^{-4}	10^{-2}	10^{-2}	10^{-2}	10^{-4}
	0.30	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}		10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.50	10^{-2}	10^{-2}	10^{-2}	10^{-2}	10^{-2}		10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
Planning Relax (182 x 12)	0.00	10^{0}	10^{-7}	10^{0}	10^{0}	10^{-3}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.05	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}		10^{-5}	10^{-5}	10^{-1}	10^{-7}	10^{-7}
	0.10	10^{0}	10^{0}	10^{0}	10^{0}	10^{0}		10^{-5}	10^{-5}	10^{-7}	10^{-7}	10^{-7}
	0.20	10^{0}	10^{-3}	10^{0}	10^{-7}	10^{-7}		10^{-5}	10^{-5}	10^{-7}	10^{-7}	10^{-7}
	0.30	10^{-7}	10^{0}	10^{-2}	10^{0}	10^{0}		10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
	0.50	10^{-3}	10^{-7}	10^{-3}	10^{-7}	10^{-7}		10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}
Fertility (100 x 9)	0.00	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-4}	10^{-4}	10^{-6}	10^{-6}	10^{-6}	10^{-6}
	0.05	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}		10^{-4}	10^{-4}	10^{-7}	10^{-7}	10^{-7}
	0.10	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}		10^{-4}	10^{-4}	10^{-7}	10^{-7}	10^{-7}
	0.20	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}		10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}
	0.30	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}		10^{-4}	10^{-7}	10^{-7}	10^{-7}	10^{-4}
	0.50	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}		10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}

Chapter 7

Conclusion and Future Work

A novel model called SPTWSVM is proposed in this project report. Compared to the original TWSVM, our proposed SPTWSVM is noise insensitive and sparse at the same time. The validity of our proposed SPTWSVM is demonstrated by numerical experiments performed on several UCI benchmark and synthetic datasets for both linear and non-linear cases. Numerical experiments clearly show that the classification accuracy of our SPTWSVM outperforms the accuracy of Sparse Pin SVM and TWSVM in most of the cases, while simultaneously maintaining sparsity and insensitivity to noise, especially, around the decision boundary. Further experiments for our second model ISPTWSVM on several UCI benchmark datasets highlight that ISPTWSVM is an efficient method for solving large scale problems with classification accuracy better or at par with the existing models. Hence, our models are excellent solvers for all varieties of binary classification problems and hold the following attractive properties:

- SPTWSVM

  Our novel SPTWSVM is insensitive to feature noise, sparse in the number of support vectors and stable for re-sampling as compared to TWSVM. It is also approximately four times faster than its Sparse Pin SVM counterpart. SPTWSVM can also be easily extended to other formulations built on top of TWSVM.

- ISPTWSVM

  Our novel ISPTWSVM introduces the principle of structural risk minimization in our SPTWSVM model, which can improve classification performance. ISPTWSVM is feasible for large scale datasets, since we bypass the calculation of inverse matrices in the dual problem which entail large time complexities ($O(m^3)$, where the matrix is of size $m \times m$). The model is also noise insensitive, sparse, stable for resampling and fast to train just like the SPTWSVM model.

Several parameters need to be regularized in both our SPTWSVM and ISPTWSVM models, and, hence, the design of proper parameter selection is our future work. Furthermore, developing more efficient training algorithms such as sequential minimal optimization (SMO) and successive over relaxation (SOR) for our models is a promising avenue of research.

Bibliography

[1] Corinna Cortes and Vladimir Vapnik. "Support-vector networks". In: *Machine learning* 20.3 (1995), pp. 273–297.

[2] Vladimir Naumovich Vapnik. "An overview of statistical learning theory". In: *IEEE transactions on neural networks* 10.5 (1999), pp. 988–999.

[3] Koby Crammer and Yoram Singer. "On the learnability and design of output codes for multiclass problems". In: *Machine learning* 47.2-3 (2002), pp. 201–233.

[4] Thomas G Dietterich and Ghulum Bakiri. "Solving multiclass learning problems via error-correcting output codes". In: *Journal of artificial intelligence research* 2 (1994), pp. 263–286.

[5] Chih-Wei Hsu and Chih-Jen Lin. "A comparison of methods for multiclass support vector machines". In: *IEEE transactions on neural networks* 13.2 (2002), pp. 415–425.

[6] John C Platt, Nello Cristianini, and John Shawe-Taylor. "Large margin DAGs for multiclass classification". In: *Advances in neural information processing systems*. 2000, pp. 547–553.

[7] Jason Weston and Chris Watkins. *Multi-class support vector machines*. Tech. rep. Citeseer, 1998.

[8] Oscar Déniz, M Castrillon, and Mario Hernández. "Face recognition using independent component analysis and support vector machines". In: *Pattern recognition letters* 24.13 (2003), pp. 2153–2157.

[9] Mathias M Adankon and Mohamed Cheriet. "Model selection for the LS-SVM. Application to handwriting recognition". In: *Pattern recognition* 42.12 (2009), pp. 3264–3270.

[10] Shutao Li et al. "Texture classification using the support vector machines". In: *Pattern recognition* 36.12 (2003), pp. 2883–2893.

[11] Thomas Navin Lal et al. "Support vector channel selection in BCI". In: *IEEE transactions on biomedical engineering* 51.6 (2004), pp. 1003–1010.

[12] Gary N Garcia Molina, Touradj Ebrahimi, and Jean-Marc Vesin. "Joint time-frequency-space classification of EEG in a brain-computer interface application". In: *EURASIP journal on applied signal processing* 2003 (2003), pp. 713–729.

[13] Giorgio Valentini, Marco Muselli, and Francesca Ruffino. "Cancer recognition with bagged ensembles of support vector machines". In: *Neurocomputing* 56 (2004), pp. 461–466.

[14] Olvi L Mangasarian and Edward W Wild. "Multisurface proximal support vector machine classification via generalized eigenvalues". In: *IEEE transactions on pattern analysis and machine intelligence* 28.1 (2006), pp. 69–74.

[15] Jayadeva, R. Khemchandani, and S. Chandra. "Twin support vector machines for pattern classification". In: *IEEE transactions on pattern analysis and machine intelligence* 29.5 (2007), pp. 905–910.

[16] Xiaolin Huang, Lei Shi, and Johan AK Suykens. "Support vector machine classifier with pinball loss". In: *IEEE transactions on pattern analysis and machine intelligence* 36.5 (2014), pp. 984–997.

[17] Shangbing Gao, Qiaolin Ye, and Ning Ye. "1-Norm least squares twin support vector machines". In: *Neurocomputing* 74.17 (2011), pp. 3590–3597.

[18] M Arun Kumar and Madan Gopal. "Application of smoothing technique on twin support vector machines". In: *Pattern recognition letters* 29.13 (2008), pp. 1842–1848.

[19] M Arun Kumar et al. "Knowledge based least squares twin support vector machines". In: *Information sciences* 180.23 (2010), pp. 4606–4618.

[20] Mohammad Tanveer. "Robust and sparse linear programming twin support vector machines". In: *Cognitive computation* 7.1 (2015), pp. 137–149.

[21] Mohammad Tanveer et al. "One norm linear programming support vector regression". In: *Neurocomputing* 173 (2016), pp. 1508–1518.

[22] Bharat Richhariya and Mohammad Tanveer. "A robust fuzzy least squares twin support vector machine for class imbalance learning". In: *Applied Soft Computing* 71 (2018), pp. 418–432.

[23] Mohammad Tanveer. "Newton method for implicit Lagrangian twin support vector machines". In: *International Journal of Machine Learning and Cybernetics* 6.6 (2015), pp. 1029–1040.

[24] Mohammad Tanveer. "Application of smoothing techniques for linear programming twin support vector machines". In: *Knowledge and Information Systems* 45.1 (2015), pp. 191–214.

[25] Mohammad Tanveer, Mohammad Asif Khan, and Shen-Shyang Ho. "Robust energy-based least squares twin support vector machines". In: *Applied Intelligence* 45.1 (2016), pp. 174–186.

[26] Yingjie Tian and Yuan Ping. "Large-scale linear nonparallel support vector machine solver". In: *Neural networks* 50 (2014), pp. 166–174.

[27] Yitian Xu and Laisheng Wang. "K-nearest neighbor-based weighted twin support vector regression". In: *Applied intelligence* 41.1 (2014), pp. 299–309.

[28] Sungmoon Cheong, Sang Hoon Oh, and Soo-Young Lee. "Support vector machines with binary tree architecture for multi-class classification". In: *Neural information processing letters and reviews* 2.3 (2004), pp. 47–51.

[29] Gjorgji Madzarov, Dejan Gjorgjevikj, and Ivan Chorbev. "A multi-class SVM classifier utilizing binary decision tree". In: *Informatica* 33.2 (2009).

[30] Yuan-Hai Shao et al. "The best separating decision tree twin support vector machine for multi-class classification". In: *Procedia Computer Science* 17 (2013), pp. 1032–1038.

[31] Yitian Xu, Zhiji Yang, and Xianli Pan. "A novel twin support-vector machine with pinball loss". In: *IEEE transactions on neural networks and learning systems* 28.2 (2017), pp. 359–370.

[32] Yuan-Hai Shao et al. "Improvements on twin support vector machines". In: *IEEE transactions on neural networks* 22.6 (2011), pp. 962–968.

[33] Craig Saunders, Alexander Gammerman, and Volodya Vovk. "Ridge regression learning algorithm in dual variables". In: (1998).

[34] Dua Dheeru and Efi Karra Taniskidou. *UCI Machine Learning Repository*. 2017. URL: http://archive.ics.uci.edu/ml.