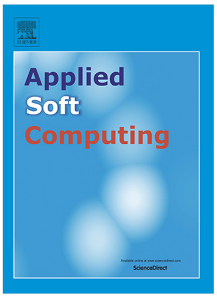


Fuzzy least squares projection twin support vector machines for class imbalance learning

M.A. Ganaie, M. Tanveer, for the Alzheimer's Disease
Neuroimaging Initiative

PII: S1568-4946(21)00855-3
DOI: https://doi.org/10.1016/j.asoc.2021.107933
Reference: ASOC 107933

To appear in: *Applied Soft Computing*

Received date : 8 April 2021
Revised date : 21 August 2021
Accepted date : 19 September 2021

Please cite this article as: M.A. Ganaie and M. Tanveer, Fuzzy least squares projection twin support vector machines for class imbalance learning, *Applied Soft Computing* (2021), doi: https://doi.org/10.1016/j.asoc.2021.107933.

Fuzzy least squares projection twin support vector machines for class imbalance learning

M.A. Ganaie[a], M. Tanveer[a,*], for the Alzheimer's Disease Neuroimaging Initiative [1]

[a]*Department of Mathematics, Indian Institute of Technology Indore, Simrol, Indore, 453552, India*

Abstract

In this paper, we propose a novel fuzzy least squares projection twin support vector machines for class imbalance learning (FLSPTSVM-CIL). Unlike twin support vector machine (TSVM) which solves two dual problems, we solve two modified primal formulations by solving two systems of linear equations. The proposed FLSPTSVM-CIL model seeks two projection directions such that the samples of two classes are well separated in the projected space. To avoid the singularity issues, we incorporate an extra regularization term to make the optimization problem positive definite. As the real world data may be imbalanced, we assign appropriate fuzzy weights to the samples such that the classifier is not biased towards the samples of the majority class. The statistical analysis and experimental results on the publicly available UCI benchmark datasets show that the proposed FLSPTSVM-CIL performs better as compared to the baseline models. To show the applications of the proposed FLSPTSVM-CIL model on real world datasets, we performed classification of Alzheimer's disease and breast cancer patients. Experimental results show that the generalization performance of the proposed FLSPTSVM-CIL model

[*] Corresponding author

Email addresses: `phd1901141006@iiti.ac.in` (M.A. Ganaie), `mtanveer@iiti.ac.in` (M. Tanveer)

[1] Data used in preparation of this article were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). As such, the investigators within the ADNI contributed to the design and implementation of ADNI and/or provided data but did not participate in analysis or writing of this report. A complete listing of ADNI investigators can be found at: http://adni.loni.usc.edu/wp-content/uploads/how_to_apply/ADNI_Acknowledgement_List.pdf

for the classification of the breast cancer patients and the mild cognitive impairment versus Alzheimer's disease subjects is better compared to the baseline models.

Keywords: Support vector machines, Twin support vector machines, Fuzzy membership, Class imbalance, Projections.

1. Introduction

Support vector machines (SVMs) [1, 2] have shown better generalization performance and hence have been applied across different domains. Unlike artificial neural networks (ANNs) which minimize the empirical risk, SVMs implement the structural risk minimization principle. As a successful supervised learning classifier, SVMs have been used in domains like detection of faces [3, 4], recognition of face expressions [5], identification of speakers [6], detection of intrusion [7], pedestrain event classification [8], brain age estimation [9] and so on. The major drawback with SVMs is the higher computational complexity. To minimize the complexity, formulations of generalized eigen values for proximal SVMs (GEPSVM) [10] was proposed. Inspired by the GEPSVM, twin support vector machine (TSVM) [11] solved two smaller size quadratic programming problems (QPPs) instead of solving one large size QPP. Computationally, TSVM is approximately four times faster than SVMs. TSVM have been efficiently used in classification [12, 13, 14, 15], regression [16, 17, 18, 19], ensemble learning [20, 21, 22] and multiclass classification [23] problems.

In class imbalance datasets, the SVM model gets biased towards majority class samples and hence shows lower generalization performance for predicting minority class samples. Multiple techniques have been used to handle imbalance problems in datasets. Fuzzy weights are assigned to the data points in fuzzy support vector machines [24] to handle class imbalance problem. Fuzzy least squares SVM [25] solves a system of equations and assigns fuzzy weights to each sample to reduce the effect of outliers. Bilateral weighted fuzzy SVM [26] and proximal bilateral weighted fuzzy SVMs [27] assume that each data point belongs to both the classes with varying fuzzy weights. Partition index is maximized in [28] while as the fuzzy SVM [29] incorporates the minimum within class scatter matrix in the fuzzy SVM formulation. Reduced universum TSVM [30] used universum concept to handle the class imbalance problem. Fuzzy concept has also been explored in prosthetic hand myoelectric-based control systems [31]. Boosting SVM [32] is proposed to handle the imbalance of classes. Oversampling technique known as synthetic minority oversampling technique (SMOTE) [33], fuzzy membership based twin SVM [34, 35] and general twin SVM with pinball loss (Pin-GTSVM) [36] were proposed either to handle the class imbalance problems or handle the noise issues.

To further improve the performance of GEPSVM, multi weight vector projection SVM (MWVSVM) [37] solved a pair of eigen value problems and seeks a weight vector such that the samples of each class are clustered around its corresponding mean and the two classes are well separated. Motivated by MWVSVM, recursive projection twin SVM (RPTSVM) [38] solved two smaller size SVM problems to project the samples such that the samples are clustered around the corresponding mean and the two classes are well separated in the projected space. Least squares recursive projection twin SVM (LSRPTSVM) [39] solved a system of linear equations which is faster compared to solving two QPPs. LSRPTSVM was extended to nonlinear case in [40] wherein the samples are projected in the kernel space to separate the data samples in the high dimensional space. In [41], the formulation extended the projection concept to multiclass problems. Comprehensive review on TSVM [42] gives the state-of-art review on the twin support vector machines based models. The projection based twin SVM models have also been used in clustering problems [43]. Unified form of fuzzy C-means and K-means algorithms and its partitional implementation [44] leads to better performance in the clustering problems.

Motivated by robust fuzzy least squares twin SVMs (RFLSTSVM) [34] and least squares recursive projection twin SVM (LSRPTSVM) [39], we propose a novel fuzzy least squares projection twin support vector machines for class imbalance learning (FLSPTSVM-CIL). In fuzzy twin support vector machines (FTWSVM), existing fuzzy membership function [45, 46, 47] based on the distance of the data from the centroid is used in TSVM model [11]. The drawbacks of the RFLSTSVM is that 1) it assumes that the matrices appearing in the dual formulation are invertible. 2) Empirical risk minimization is implemented which results in lower generalization performance. Similarly, the drawback of LSRPTSVM model is that it suffers in class imbalance problems. The TSVM, LSTSVM and LSRPTSVM models may be biased towards the majority class samples which results in lower generalization performance. Also, TSVM and FTWSVM models solve a pair of quadratic programming problem (QPP) which is slower. To overcome these drawbacks, we propose a novel FLSPTSVM-CIL in which matrices in the primal formulation are positive definite and solve the linear system of equations such that the samples of each class are clustered around its corresponding mean and as far as possible from the samples of the other class. The proposed FLSPTSVM-CIL model doesn't require any optimization toolbox. For computational efficiency, only one projection axis is generated for each class in the proposed FLSPTSVM-CIL model. The advantages of the proposed FLSPTSVM-CIL are:

- Unlike RFLSTSVM, the proposed FLSPTSVM-CIL incorporates an extra regularisation in each objective function to maximise the margin. Also, structural risk minimization principle is implemented in the proposed FLSPTSVM-CIL model which is the marrow of statistical learning.
- The proposed FLSPTSVM-CIL model seeks projections such that the samples of each class are clustered around its corresponding mean and the samples of different classes are as far as possible.
- Unlike LSRPTSVM, the proposed FLSPTSVM-CIL assigns fuzzy weights to each samples to reduce the effect of class imbalance.
- Experimental results show the efficacy of the proposed FLSPTSVM-CIL compared to the baseline models.

The general outline of this paper: Section 1 gives introduction, related work is given in Section 2, and Section 3 discusses the proposed work. Evaluation of experimental results is given in Section 4 and conclusion with future remarks are given in Section 5.

In this paper, all vectors are assumed as column vectors unless transformed to row vector. Let $A \in \mathbb{R}^{m_1 \times n}$ and $B \in \mathbb{R}^{m_2 \times n}$ be the samples of the positive and negative classes, respectively, of a binary classification problems. Here, we assume that the class with minority samples is treated as positive class while as the class with majority samples is treated as negative class. Each sample x belonging to either of the classes is of the form $x \in \mathbb{R}^n$ where n is the feature dimension. Imbalance ratio (IR) is defined as

$$IR = \frac{\text{Number of samples in negative class}}{\text{Number of samples in positive class}}. \tag{1}$$

2. Related Work

Here, we briefly discuss the formulations of baseline methods: twin SVM (TSVM), least squares TSVM (LSTSVM), fuzzy TSVMs (FTWSVM) and robust fuzzy LSTSVMs (RFLSTSVM).

2.1. TSVM

The objective function of the TSVM [11] for nonlinear case is expressed as follows:

$$\begin{aligned} &\underset{u_1,b_1}{min}\ \ \frac{1}{2}\|K(A,C^t)u_1+e_1b_1\|^2+c_1e_2^t\xi_1 \\ &s.t.\ \ -(K(B,C^t)u_1+e_2b_1)+\xi_1\geq e_2,\ \xi_1\geq 0 \end{aligned} \tag{2}$$

and

$$\begin{aligned} &\underset{u_2,b_2}{min}\ \ \frac{1}{2}\|K(B,C^t)u_2+e_2b_2\|^2+c_2e_1^t\xi_2 \\ &s.t.\ \ (K(A,C^t)u_2+e_1b_2)+\xi_2\geq e_1,\ \xi_2\geq 0, \end{aligned} \tag{3}$$

where $[u_1;b_1]$ and $[u_2;b_2]$ are the hyperplanes proximal to positive and negative classes, respectively, ξ_i and e_i are the slack variables and vector of ones, respectively, for $i=1,2$.

The dual problems of (2) and (3), with α and β as Lagrange multipliers, are given as follows:

$$\begin{aligned} &\underset{\alpha}{Max}\ \ e_2^t\alpha-\frac{1}{2}\alpha^tF(E^tE)^{-1}F^t\alpha \qquad (4) \\ &s.t.\ \ 0\leq\alpha\leq c_1e_2 \end{aligned}$$

and

$$\begin{aligned} &\underset{\beta}{Max}\ \ e_1^t\beta-\frac{1}{2}\beta^tE(F^tF)^{-1}E^t\beta \qquad (5) \\ &s.t.\ \ 0\leq\beta\leq c_2e_1, \end{aligned}$$

where $F=[K(B,C^t)\quad e_2]$, $E=[K(A,C^t)\quad e_1]$ and $C=[A;B]$. Solving (4) and (5) leads to following hyperplanes:

$$\begin{bmatrix}u_1\\b_1\end{bmatrix}=-(E^tE+\delta I)^{-1}F^t\alpha, \tag{6}$$

$$\begin{bmatrix}u_2\\b_2\end{bmatrix}=(F^tF+\delta I)^{-1}E^t\beta, \tag{7}$$

where a small positive number δ avoids the ill conditioning of the matrices and I is the identity matrix of appropriate dimensions.

2.2. LSTSVM

Least squares twin SVM (LSTSVM) [48] solves a system of linear equations which is faster as compared to solving a QPP.

The primal problem of LSTSVM model is expressed as follows:

$$\begin{aligned} \underset{u_1,b_1}{min} \quad & \frac{1}{2}\|K(A,C^t)u_1+e_1b_1\|^2+\frac{c_1}{2}\xi_1^t\xi_1 \\ s.t. \quad & -(K(B,C^t)u_1+e_2b_1)+\xi_1=e_2 \end{aligned} \tag{8}$$

and

$$\begin{aligned} \underset{u_2,b_2}{min} \quad & \frac{1}{2}\|K(B,C^t)u_2+e_2b_2\|^2+\frac{c_2}{2}\xi_2^t\xi_2 \\ s.t. \quad & K(A,C^t)u_2+e_1b_2+\xi_2=e_1. \end{aligned} \tag{9}$$

Using the constraints in the corresponding formulation, we get

$$\underset{u_1,b_1}{min} \quad \frac{1}{2}\|(K(A,C^t)u_1+e_1b_1)\|^2+\frac{c_1}{2}\|K(B,C^t)u_1+e_2b_1+e_2\|^2. \tag{10}$$

Taking the gradient of QPP (10) w.r.t. u_1 and b_1 and setting it to zero, we get

$$\begin{bmatrix} u_1 \\ b_1 \end{bmatrix} = -(F^tF+\frac{1}{c_1}E^tE)^{-1}F^te_2. \tag{11}$$

Similarly, substitute the constraints in the corresponding formulation of (9) and take the gradient of QPP w.r.t. u_2 and b_2 and setting it to zero, we get

$$\begin{bmatrix} u_2 \\ b_2 \end{bmatrix} = (E^tE+\frac{1}{c_2}F^tF)^{-1}E^te_1. \tag{12}$$

2.3. FTWSVM

Distance based fuzzy membership weights [45, 46, 47] are incorporated in the formulation of TSVM [11]. The objective functions of the FTWSVM are given as

$$\begin{aligned} &\underset{u_1,b_1}{min} \ \ \frac{1}{2}\|K(A,C^t)u_1 + e_1b_1\|^2 + c_1 s_2^t \xi_1 \\ &s.t. \ \ -(K(B,C^t)u_1 + e_2 b_1) + \xi_1 \geq e_2, \xi_1 \geq 0 \end{aligned} \tag{13}$$

and

$$\begin{aligned} &\underset{u_2,b_2}{min} \ \ \frac{1}{2}\|K(B,C^t)u_2 + e_2b_2\|^2 + c_2 s_1^t \xi_2 \\ &s.t. \ \ K(A,C^t)u_2 + e_1 b_2 + \xi_2 \geq e_1, \xi_2 \geq 0, \end{aligned} \tag{14}$$

where ξ_i represent the slack variables, c_i represents the penalty parameters and s_i represent the fuzzy membership weights. Here, distance of a sample from the class centroid is used as the fuzzy membership weight. Let d denote how far the sample is from the class centroid and δ be a small positive integer then the fuzzy membership function is given as:

$$fmem = 1 - \frac{d}{max(d) + \delta}. \tag{15}$$

Similar to TSVM, the dual of objective functions (13) and (14) is given as

$$\begin{aligned} &\underset{\alpha}{min} \ \ \frac{1}{2}\alpha^t F(E^t E)^{-1} F^t \alpha - e_2^t \alpha \\ &s.t. \ \ 0 \leq \alpha \leq s_2 c_1 \end{aligned} \tag{16}$$

and

$$\begin{aligned} &\underset{\beta}{min} \ \ \frac{1}{2}\beta^t E(F^t F)^{-1} E^t \beta - e_1^t \beta \\ &s.t. \ \ 0 \leq \beta \leq s_1 c_2. \end{aligned} \tag{17}$$

The optimal hyperplanes are given as

$$\begin{bmatrix} u_1 \\ b_1 \end{bmatrix} = -(E^t E + \delta I)^{-1} F^t \alpha \tag{18}$$

$$\text{and} \begin{bmatrix} u_2 \\ b_2 \end{bmatrix} = (F^t F + \delta I)^{-1} E^t \beta, \tag{19}$$

where a small value δ avoids the ill conditioning of the matrices.

2.4. RFLSTSVM

The formulation of robust fuzzy least squares twin SVM (RFLSTSVM) [34] is given as:

$$\begin{aligned} \underset{u_1,b_1}{min} \quad & \frac{1}{2}\|K(A,C^t)u_1 + e_1 b_1\|^2 + \frac{c_1}{2}(s_2\xi_1)^t(s_2\xi_1) \\ s.t. \quad & -(K(B,C^t)u_1 + e_2 b_1) + \xi_1 = e_2, \end{aligned} \tag{20}$$

and

$$\begin{aligned} \underset{u_2,b_2}{min} \quad & \frac{1}{2}\|K(B,C^t)u_2 + e_2 b_2\|^2 + \frac{c_2}{2}(s_1\xi_2)^t(s_1\xi_2) \\ s.t. \quad & K(A,C^t)u_2 + e_1 b_2 + \xi_2 = e_1, \end{aligned} \tag{21}$$

where s_1 and s_2 denote fuzzy membership functions and ξ_1, ξ_2 are the slack variables.

The function for assigning the fuzzy weights to the samples is given as

$$fuzzy\_mem = \begin{cases} 1, & \text{for positive class samples,} \\ z + z(\frac{exp(c_0(\frac{d_1-d_2}{d} - \frac{d_2}{r_2}) - exp(-2c_0))}{exp(c_0) - exp(-2c_0)}), & \text{otherwise.} \end{cases} \tag{22}$$

Here, $z = \frac{1}{1+IR}$, IR is imbalance ratio, d_1 is the distance of the sample from the positive class centroid and d_2 is the distance of the sample from the negative class centroid, distance between positive class and negative class centroid is d, and negative class maximum distance from centroid is given by r_2, c_0 is the exponential scale of the membership function.

Following the similar approach as in LSTSVM for solving the QPPs (20) and (21), the optimal hyperplanes are given as

$$\begin{bmatrix} u_1 \\ b_1 \end{bmatrix} = -(T^t T + \frac{1}{c_1} R^t R)^{-1} T^t s_2 e_2, \tag{23}$$

where $R = [A \;\; e_1]$ and $T = [s_2 B \;\; s_2 e_2]$ and

$$\begin{bmatrix} u_2 \\ b_2 \end{bmatrix} = (R^T R + \frac{1}{c_2} T^t T)^{-1} R^t s_1 e_1, \tag{24}$$

where $R = [s_1 A \;\; s_1 e_1]$ and $T = [B \;\; e_2]$.

3. Proposed fuzzy least squares projection twin support vector machines for class imbalance learning (FLSPTSVM-CIL)

In class imbalance learning, due to majority of the samples of a particular class, the classifier gets biased towards the majority class and results in misclassifications. To reduce the effect of majority class samples, the proposed FLSPTSVM-CIL assigns fuzzy weights to the data samples. The fuzzy weights are assigned to the data samples using fuzzy membership function given in equation (22) which uses IR information to improve the model performance. To make the model more robust, the proposed FLSPTSVM-CIL implements the structural risk minimization principle and seeks the projection axis to minimise the within class variance of the projected samples. The proposed FLSPTSVM-CIL solves a system of equations and no explicit toolbox is needed to solve the optimization problem.

The formulation of proposed FLSPTSVM-CIL model for both linear and non-linear cases is given as follows:

3.1. Linear FLSPTSVM-CIL

The formulation of the proposed FLSPTSVM-CIL for linear case is given as:

$$\begin{aligned} \min_{u_1,b_1} \ & \frac{1}{2}(\|V_1^{\frac{1}{2}}u_1\|^2 + b_1^2) + \frac{c_3}{2}(\|u_1\|^2 + b_1^2) + \frac{1}{2}\eta_1^t\eta_1 \\ & \qquad\qquad + \frac{c_1}{2}(s_2\xi_2)^t(s_2\xi_2) \\ s.t. \ \ & Au_1 + e_2b_1 = \eta_1, \\ & -(Bu_1 + e_1b_1) + \xi_2 = e_1 \end{aligned} \tag{25}$$

and

$$\begin{aligned} \min_{u_2,b_2} \ & \frac{1}{2}(\|V_2^{\frac{1}{2}}u_2\|^2 + b_2^2) + \frac{c_4}{2}(\|u_2\|^2 + b_2^2) + \frac{1}{2}\eta_2^t\eta_2 \\ & \qquad\qquad + \frac{c_2}{2}(s_1\xi_1)^t(s_1\xi_1) \\ s.t. \ \ & Bu_2 + e_1b_2 = \eta_2, \\ & (Au_2 + e_2b_2) + \xi_1 = e_2, \end{aligned} \tag{26}$$

where the positive and negative class samples are given by A and B, respectively. V_1 is the class variance of positive class and V_2 is the class variance of negative class. Mathematically, the variance of each class is given by

$$V_1 = \sum_{i=1}^{m_1} \Big(x_i^{(1)} - \frac{1}{m_1}\sum_{j=1}^{m_1} x_j^{(1)}\Big)\Big(x_i^{(1)} - \frac{1}{m_1}\sum_{j=1}^{m_1} x_j^{(1)}\Big)^t \tag{27}$$

and

$$V_2 = \sum_{i=1}^{m_2} \Big(x_i^{(2)} - \frac{1}{m_2}\sum_{j=1}^{m_2} x_j^{(2)}\Big)\Big(x_i^{(2)} - \frac{1}{m_2}\sum_{j=1}^{m_2} x_j^{(2)}\Big)^t, \tag{28}$$

where, $x_i^{(1)} \in A$ and $x_i^{(2)} \in B$.

The first term in the optimization problem (25) seeks projections such that the samples of a class are clustered around its corresponding mean, the second term implements the structural risk minimization principle, the third term makes hyperplane proximal to the corresponding class and the last term minimizes the error of the fuzzy weighted samples. Fuzzy weights are assigned to data samples to reduce the effect of class imbalance, such that the hyperplane is not biased towards the majority class samples. With the similar objectives, the optimization problem (26) is designed to obtain the second optimal hyperplane.

Using the constraints of (25) in its objective function, we get

$$\begin{aligned} \min_{u_1,b_1}\ & \frac{1}{2}(\|V_1^{\frac{1}{2}}u_1\|^2 + b_1^2) + \frac{c_3}{2}(\|u_1\|^2 + b_1^2) + \frac{1}{2}\|Au_1 + e_2b_1\|^2 \\ & + \frac{c_1}{2}\|s_2(Bu_1 + e_1b_1 + e_1)\|^2. \end{aligned} \tag{29}$$

Take the gradient of (29) w.r.t. u_1 and b_1 and set them to zero, we get

$$\begin{bmatrix} u_1 \\ b_1 \end{bmatrix} = -\Big(\hat{V_1} + H^tH + c_3I + c_1G^tG\Big)^{-1} \times c_1(G^ts_2e_1). \tag{30}$$

where $H = [A \ \ e_2]$, $G = [s_2B \ \ s_2e_1]$ and $\hat{V_1} = \begin{bmatrix} V_1 & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}$, here, $\mathbf{0}$ is the vector of zeros with appropriate dimensions.

Similarly, the solution of (26) is given as:

$$\begin{bmatrix} u_2 \\ b_2 \end{bmatrix} = \Big(\hat{V_2} + R^tR + c_4I + c_2T^tT\Big)^{-1} \times c_2(R^ts_1e_2), \tag{31}$$

Algorithm 1 Linear FLSPTSVM-CIL.

```
Input: $A \in \mathbb{R}^{m_1 \times n}$, $B \in \mathbb{R}^{m_2 \times n}$.
Calculate the fuzzy weights using the fuzzy membership function (22).
Calculate the variance of the positive and negative class using (27) and (28), respectively.
Select the best parameters by using the grid search method.
Determine $(u_1, b_1)$, and $(u_2, b_2)$ by solving the equations (30) and (31), respectively.
For classifying testing point $x_i$, assign the class label using function (32).
```

where $R = [s_1 A \quad s_1 e_2]$, $\quad T = [B \quad e_1]$ and $\hat{V_2} = \begin{bmatrix} V_2 & \mathbf{0} \\ \mathbf{0} & 1 \end{bmatrix}$, here, $\mathbf{0}$ is the vector of zeros with appropriate dimensions.

Note that addition of the regularization term makes the matrices $\Big(\hat{V_1} + H^t H + c_3 I + c_1 G^t G\Big)$ and $\Big(\hat{V_2} + R^t R + c_4 I + c_2 T^t T\Big)$ as positive definite.

The class of a test sample $x \in \mathbb{R}^n$ is given as follows:

$$Class(x) = \underset{i=1,2}{argmin} \quad \Big|\Big((u_i^t x - u_i^t \frac{1}{m_i} \sum_{j=1}^{m_i} x_j^{(i)}\Big) + b_i\Big|. \tag{32}$$

The linear proposed FLSPTSVM-CIL is summarized in Algorithm 1.

3.2. Non-Linear FLSPTSVM-CIL

The formulation of the proposed FLSPTSVM-CIL for non-linear case is given as:

$$\begin{aligned} \min_{u_1, b_1} \quad & \frac{1}{2}(\|V_1^{\frac{1}{2}} u_1\|^2 + b_1^2) + \frac{c_3}{2}(\|u_1\|^2 + b_1^2) + \frac{1}{2}\eta_1^t \eta_1 \\ & \qquad\qquad\qquad + \frac{c_1}{2}(s_2 \xi_2)^t (s_2 \xi_2) \\ s.t. \quad & K(A, C^t) u_1 + e_2 b_1 = \eta_1, \\ & -(K(B, C^t) u_1 + e_1 b_1) + \xi_2 = e_1 \end{aligned} \tag{33}$$

and

$$\begin{aligned} \underset{u_2,b_2}{min} \quad & \frac{1}{2}(\|V_2^{\frac{1}{2}}u_2\|^2 + b_2^2) + \frac{c_4}{2}(\|u_2\|^2 + b_2^2) + \frac{1}{2}\eta_2^t\eta_2 \\ & \qquad\qquad\qquad\qquad + \frac{c_2}{2}(s_1\xi_1)^t(s_1\xi_1) \\ s.t. \quad & K(B,C^t)u_2 + e_1b_2 = \eta_2, \\ & (K(A,C^t)u_2 + e_2b_2) + \xi_1 = e_2, \end{aligned} \tag{34}$$

where, the positive and negative class samples are given by A and B, respectively. V_1 and V_2 are the class variances of the positive and negative classes respectively. Also, $C = [A; B], K(A,C^t)$ and $K(B,C^t)$ are the non-linear transformation in the higher dimensional space.

Mathematically, the variance of each class is given by

$$V_1 = \sum_{i=1}^{m_1}\Big(x_i^{(1)} - \frac{1}{m_1}\sum_{j=1}^{m_1}x_j^{(1)}\Big)\Big(x_i^{(1)} - \frac{1}{m_1}\sum_{j=1}^{m_1}x_j^{(1)}\Big)^t \tag{35}$$

and

$$V_2 = \sum_{i=1}^{m_2}\Big(x_i^{(2)} - \frac{1}{m_2}\sum_{i=1}^{m_2}x_i^{(2)}\Big)\Big(x_i^{(2)} - \frac{1}{m_2}\sum_{i=1}^{m_2}x_i^{(2)}\Big)^t, \tag{36}$$

where, $x_i^{(1)} \in K(A,C^t)$ and $x_i^{(2)} \in K(B,C^t)$.

Similar to linear case, we can obtain the following

$$\begin{bmatrix}u_1\\b_1\end{bmatrix} = -\Big(\hat{V}_1 + H^tH + c_3I + c_1G^tG\Big)^{-1} \times c_1(G^ts_2e_1). \tag{37}$$

where, $H = [K(A,C^t) \;\; e_2]$, $\;G = [s_2K(B,C^t) \;\; s_2e_1]$ and $\hat{V}_1 = \begin{bmatrix}V_1 & \mathbf{0}\\ \mathbf{0} & 1\end{bmatrix}$ here, $\mathbf{0}$ is the vector of zeros with appropriate dimensions.

Similarly, the solution of QPP (34) is given as:

$$\begin{bmatrix}u_2\\b_2\end{bmatrix} = \Big(\hat{V}_2 + R^tR + c_4I + c_2T^tT\Big)^{-1} \times c_2(R^ts_1e_2), \tag{38}$$

where, $R = [s_1K(A,C^t) \;\; s_1e_2]$, $\;T = [K(B,C^t) \;\; e_1]$ and $\hat{V}_2 = \begin{bmatrix}V_2 & \mathbf{0}\\ \mathbf{0} & 1\end{bmatrix}$, here, $\mathbf{0}$ is the vector of zeros with appropriate dimensions.

Algorithm 2 Non-Linear FLSPTSVM-CIL.

```
Input: $A \in \mathbb{R}^{m_1 \times n}$, $B \in \mathbb{R}^{m_2 \times n}$.
Calculate the fuzzy weights using the fuzzy membership function (22).
Calculate the variance of the positive and negative class using (35) and (36), respectively.
Select an appropriate kernel function $K$.
Select the best parameters by using the grid search method.
Determine $(u_1, b_1)$, and $(u_2, b_2)$ by solving the equations (37) and (38), respectively.
For classifying testing point $x_i$, assign the class label using function (39).
```

In both linear and non-linear cases of the proposed FLSPTSVM-CIL, the data samples are weighted via fuzzy membership function given in (22).

The class of a test sample $x \in \mathbb{R}^n$ is given as follows:

$$Class(x) = \underset{i=1,2}{argmin} \quad \left| \left(u_i^t K(x, C^t) - u_i^t \frac{1}{m_i} \sum_{j=1}^{m_i} K(x_j^{(i)}, C^t) \right) + b_i \right|. \tag{39}$$

Note that addition of the regularization term makes the matrices $\left(\hat{V}_1 + H^t H + c_3 I + c_1 G^t G\right)$ and $\left(\hat{V}_2 + R^t R + c_4 I + c_2 T^t T\right)$ as positive definite, hence, the proposed FLSPTSVM-CIL shows more generalization and is more stable as compared to RFLSTSVM and LSTSVM methods.

The non-linear proposed FLSPTSVM-CIL is summarized in Algorithm 2.

4. Experimental Results

In this section, we discuss the experimental setup used and specify the range of hyperparameters used for the classification models. Also, we discuss the different performance measures and statistical tests used to evaluate the classification models. The performance of the models is evaluated on the UCI and KEEL benchmark datasets and biological datasets. The biological datasets included in this study are Alzheimer's disease and Breast cancer subjects.

All the experiments were conducted on Windows-10 with 128-GB RAM Intel(R) Xeon(R) CPU E5-2697 v4 2.30GHz with MATLAB R2017b. We used Gaussian kernel, $K(x, y) = exp^{-(\|x-y\|^2/\mu^2)}$ where μ is a hyperparameter. We divided the data randomly into $70 : 30$ ratio for training and testing sets for the training and evaluation of the models respectively. The optimal parameters corresponding to the different classification models were obtained from the ranges $c_0 = [0.5, 1, 1.5, 2, 2.5], \mu = c_i = [10^{-5}, 10^{-4}, \ldots, 10^4, 10^5]$ where $i = 1, 2, 3, 4$. We used 5-fold cross validation on the training data to obtain the optimal parameters of different models via grid search approach. In 5-fold cross-validation, the data sets are separated into five equally-sized disjoint subsets, and the training of classifier is performed on all the subsets except with the exception of one, which is called test data. The performance of the classification models on the testing set using optimal parameters is reported as the classification accuracy.

The performance of the baseline models (here, TSVM, LSTSVM, FTWSVM and RFLSTSVM-CIL) and the proposed FLSPTSVM-CIL is evaluated with area under the curve (AUC) or accuracy which is defined as:

$$\text{AUC or Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{40}$$

$$\text{Sensitivity} = \frac{TP}{TP + FN}, \tag{41}$$

$$\text{Specificity} = \frac{TN}{TN + FP}, \tag{42}$$

where, TP, TN, FP, FN are the true positive, true negative, false positive and false negative, respectively.

To evaluate the models statistically, we use Friedman test. In Friedman test, each model is assigned a rank on each dataset such that the lower rank is assigned to the model which achieves better accuracy on a given dataset. Hence, lower the rank of the model the better performance it shows. Under Null hypothesis, all models are performing equally and hence their average ranks are equal, the Friedman statistics are distributed according to χ_F^2 with $(k - 1)$ degrees of freedom as:

$$\chi_F^2 = \frac{12N}{k(k+1)}\Big[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\Big], \tag{43}$$

$$F_F = \frac{(N-1)\chi_F^2}{N(k-1) - \chi_F^2}, \tag{44}$$

where $R_j = \frac{1}{N}\sum_i r_i^j$ and r_i^j denotes the rank of the j^{th} classification model on the i^{th} dataset. Here, N is the total number of datasets on which the k

classification models are evaluated. F_F is distributed with $(k-1)(N-1)$ degrees of freedom.

4.1. UCI [49] and KEEL [50] benchmark datasets

The performance of the models is evaluated on binary class datasets from UCI [49] and KEEL [50] benchmark repositories. The details of the datasets are given in Table-1. The first column gives the dataset name, second column represents the training set size, third column represents the testing size, IR-all represents the imbalance ratio in the whole dataset and IR-train represents the imbalance ratio in the training set.

Table-2 gives the performance of the baseline models and the proposed FLSPTSVM-CIL model on the given datasets. One can see that the proposed FLSPTSVM-CIL model achieves better average accuracy than the baseline models. The accuracy of proposed FLSPTSVM-CIL on abalone9-18 dataset is atleast 9% better than the TSVM, LSTSVM, FTWSVM and RFLSTSVM-CIL models. On ecoli-0-1-4-7_vs_2-3-5-6, the accuracy of proposed FLSPTSVM-CIL is atleast 3% better than TSVM, 8% better than LSTSVM and FTWSVM, 9% better than the RFLSTSVM-CIL models. Similarly, in glass5 dataset the proposed FLSPTSVM-CIL achieved 93% accuracy followed by RFLSTSVM-CIL with 85%, TSVM and FTWSVM with 75% and LSTSVM with 71% accuracy. TSVM and proposed FLSPTSVM-CIL model achieved 100% accuracy on new-thyroid1 dataset. The proposed FLSPTSVM-CIL model achieved 100% accuracy on shuttle-6_vs_2-3 dataset while as other baseline models achieved 75% accuracy.

The average rank of each model is given in Table-2. The average rank of the TSVM, LSTSVM, FTWSVM, RFLSTSVM-CIL, and proposed FLSPTSVM-CIL are $2.7759, 3.7759, 3.2586, 3,$ and 2.1897, respectively. At 5% level of significance, the critical values of $F(4,112) = 2.445$. After simple calculations, we get $\chi_F^2 = 15.9651$ and $F_F = 4.4687$. Hence, we reject the null hypothesis. Thus, significant difference exists among the given models. To analyze the difference among the models, we use pairwise post-hoc Nemenyi test. With Nemenyi test, the critical difference at 5% level of significance, $CD = q_\alpha\sqrt{\frac{k(k+1)}{6N}} = 2.7280\sqrt{\frac{5\times6}{6\times29}} = 1.1327$. One can see that the proposed FLSPTSVM-CIL model is statistically better than the LSTSVM model. However, Nemenyi test fails to detect the significant difference among the proposed FLSPTSVM-CIL model and other models except LSTSVM model. From Table-2, it is clear that the proposed FLSPTSVM-CIL model achieved better average accuracy and lower average rank as compared to the given baseline models. To further evaluate the models, we used overall win-tie-loss test. One can see from Table-2 that the proposed FLSPTSVM-CIL emerged as the overall winner in 11 datasets followed by RFLSTSVM-CIL with 5 and FTWSVM with 4 wins. TSVM and LSTSVM models achieved 2 overall wins. Hence, the performance of the proposed FLSPTSVM-CIL is better as compared to the baseline models.

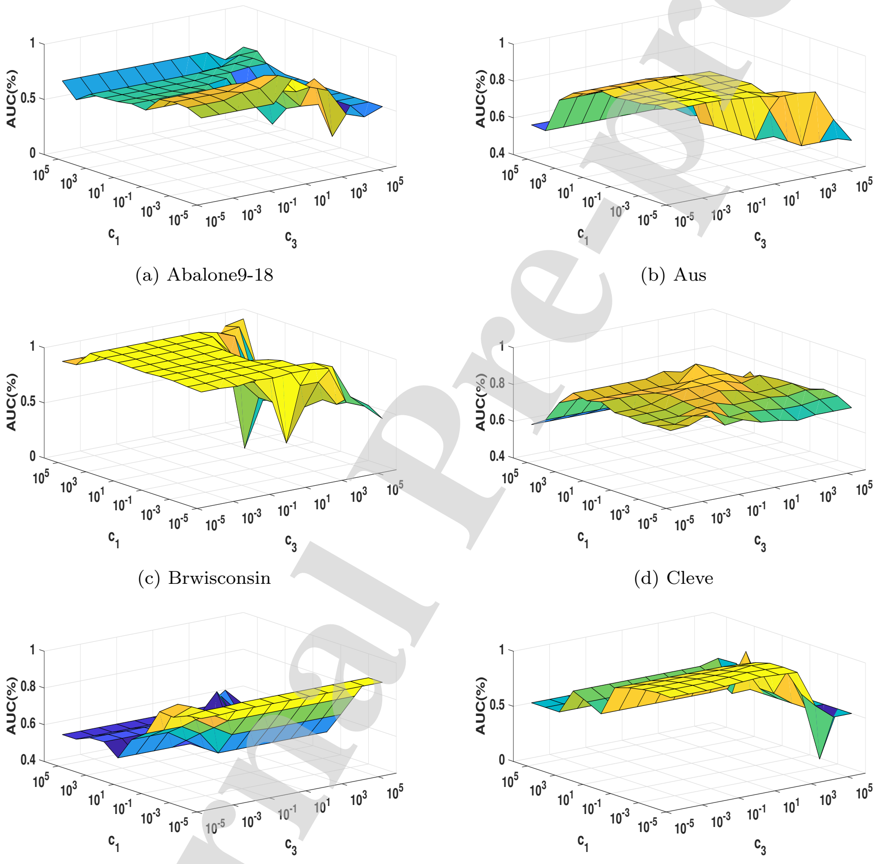
To evaluate the sensitivity of the proposed FLSPTSVM-CIL model with the varying hyperparameters c_1 and c_3, the Figures-1a to Figure-1f are plotted. One can see that to obtain the better performance, we need to choose the hyperparameters carefully.

4.2. Experiments on real-world biomedical images

Here, we evaluate the performance of the models in real world biomedical benchmark datasets. We evaluated the models for classification of Alzheimer's disease subjects and breast cancer subjects.

4.2.1. Classification of Alzheimer's disease subjects

The images of Alzheimer's disease (AD) subjects were downloaded from Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). In 2003, ADNI as a public–private partnership was launched, with Michael W. Weiner as the Principal Investigator, with the objective of analyzing the neuroimaging techniques like magnetic resonance imaging (MRI), positron emission tomography (PET), other biological markers, and clinical neuropsychological tests for estimation of the onset of Alzheimer's disease from the state of mild cognitive impairment. For further details, refer to www.adni-info.org. For the classification of AD subjects, we used features obtained with Volume based morphometry (VolBM). For extracting the features, we followed the same pipeline as given in [51]. The performance of the proposed FLSPTSVM-CIL model and the baseline models is evaluated in terms of the classification of the control normal subjects versus Alzheimer's disease subjects (CN vs AD), control normal subjects versus mild cognitive impairment subjects (CN vs MCI), and mild cognitive impairment subjects versus Alzheimer's disease subjects (MCI vs AD). We evaluated the performance of the proposed FLSPTSVM-CIL and the baseline models at varying level of feature percentage. The details of features and the samples of each set used for training and testing is given in Table-3.



(a) Abalone9-18

(b) Aus

(c) Brwisconsin

(d) Cleve

(e) Ecoli-0-1-4-7_vs_5-6

(f) Ecoli4

Figure 1: The sensitivity of the proposed FLSPTSVM-CIL model with the varying hyper-parameters c_1 and c_3.

Table 1: The details of the UCI benchmark datasets.

Datasets	Train	Test	IR-All	IR-Train
abalone9-18	(510×8)	(221×8)	16.4	16
aus	(481×15)	(209×15)	1.25	1.26
brwisconsin	(477×10)	(206×10)	0.54	0.69
bupa-or-liver-disorders	(240×7)	(105×7)	1.38	1.31
checkerboard_Data	(481×15)	(209×15)	1.25	1.26
cleve	(206×14)	(91×14)	1.17	1.12
cmc	(1030×10)	(443×10)	0.75	0.76
ecoli-0-1-4-7_vs_2-3-5-6	(234×8)	(102×8)	10.59	11.32
ecoli-0-1-4-7_vs_5-6	(231×7)	(101×7)	12.28	12.59
ecoli-0-1_vs_2-3-5	(169×8)	(75×8)	9.17	8.94
ecoli-0-1_vs_5	(167×7)	(73×7)	11	10.93
ecoli-0-2-3-4_vs_5	(140×8)	(62×8)	9.1	8.33
ecoli-0-4-6_vs_5	(141×7)	(62×7)	9.15	9.85
ecoli-0-6-7_vs_5	(153×7)	(67×7)	10	9.93
ecoli4	(234×8)	(102×8)	15.8	14.6
glass5	(148×10)	(66×10)	22.78	20.14
heart-stat	(188×14)	(82×14)	0.8	0.83
iono	(244×34)	(107×34)	0.56	0.59
monk2	(419×8)	(182×8)	1.92	1.95
new-thyroid1	(149×6)	(66×6)	5.14	4.73
ripley	(874×3)	(376×3)	1	1.01
segment0	(1614×20)	(694×20)	6.02	6.05
shuttle-6_vs_2-3	(160×10)	(70×10)	22	25.67
shuttle-c0-vs-c4	(1279×10)	(550×10)	13.87	13.87
sonar	(144×61)	(64×61)	1.14	1.12
transfusion	(522×5)	(226×5)	3.2	2.9
vehicle1	(591×19)	(255×19)	2.9	3.1
vehicle2	(591×19)	(255×19)	2.88	2.79
vowel	(690×11)	(298×11)	9.98	9.95

Table 2: The classification performance of the models on UCI benchmark datasets with Gaussian kernel.

Datasets	TSVM AUC $(c_1 = c_2, \mu)$	LSTSVM AUC $(c_1 = c_2, \mu)$	FTWSVM AUC $(c_1 = c_2, \mu)$	RFLSTSVM-CIL AUC $(c_0, c_1 = c_2, \mu)$	Proposed FLSPTSVM-CIL AUC $(c_0, c_1 = c_2, c_3 = c_4, \mu)$
abalone9-18	0.8295 $(10^2, 1)$	0.7083 $(1, 0.1)$	0.8046 $(10^{-2}, 1)$	0.7117 $(2, 0.1, 0.1)$	0.9105 $(1.5, 1, 10^{-5}, 10)$
aus	0.8516 $(0.1, 10^2)$	0.7191 $(10^{-4}, 10^4)$	0.8512 $(10^{-5}, 10^4)$	0.8458 $(0.5, 1, 10^4)$	0.8623 $(1.5, 1, 0.1, 10^3)$
brwisconsin	0.9685 $(10^{-3}, 10^2)$	0.9858 $(10^{-2}, 10)$	0.9938 $(10^{-5}, 10^2)$	0.9907 $(0.5, 10^{-2}, 10^5)$	0.9876 $(0.5, 0.1, 0.1, 10^3)$
bupa-or-liver-disorders	0.6833 $(0.1, 10^4)$	0.7111 $(1, 10^4)$	0.7111 $(1, 10^4)$	0.6965 $(2, 1, 10^4)$	0.7121 $(1, 1, 10^{-4}, 10^4)$
checkerboard_Data	0.8516 $(0.1, 10^2)$	0.7191 $(10^{-4}, 10^4)$	0.8512 $(10^{-5}, 10^4)$	0.8458 $(0.5, 1, 10^4)$	0.8623 $(1.5, 1, 0.1, 10^3)$
cleve	0.8287 $(10^{-5}, 10^4)$	0.8483 $(10^{-3}, 10^3)$	0.8162 $(10^{-5}, 10^4)$	0.8118 $(2.5, 0.1, 10^4)$	0.8118 $(0.5, 10^{-5}, 1, 10)$
cmc	0.6892 $(10^{-2}, 10^2)$	0.6776 $(1, 10^2)$	0.7061 $(10^{-5}, 10^2)$	0.6977 $(0.5, 1, 10^2)$	0.7202 $(2.5, 10^{-3}, 10^{-4}, 10^2)$
ecoli-0-1-4-7_vs_2-3-5-6	0.9 $(1, 10^4)$	0.85 $(10, 10^3)$	0.85 $(0.1, 10^4)$	0.8446 $(1, 10^{-2}, 10^4)$	0.9391 $(2, 10^{-5}, 10^{-4}, 10^2)$
ecoli-0-1-4-7_vs_5-6	0.875 $(10, 10^5)$	0.875 $(1, 10^4)$	0.8125 $(1, 10^4)$	0.8159 $(1.5, 10^{-2}, 10^5)$	0.8535 $(1.5, 10^{-5}, 10^{-4}, 10^2)$
ecoli-0-1_vs_2-3-5	0.7143 $(0.1, 10^4)$	0.7143 $(1, 10^3)$	0.6429 $(10^{-5}, 10^3)$	0.7784 $(1.5, 10^{-5}, 10^3)$	0.8424 $(1, 10^3, 10^4, 10^3)$
ecoli-0-1_vs_5	0.8333 $(1, 10^5)$	0.8259 $(1, 10^4)$	0.6667 $(0.1, 10^5)$	0.8794 $(1.5, 10^{-2}, 10^4)$	0.9017 $(0.5, 10^{-4}, 10^{-2}, 10^2)$
ecoli-0-2-3-4_vs_5	0.9737 $(10, 10^4)$	0.9912 $(0.1, 10^4)$	1 $(10^{-3}, 10^4)$	0.9825 $(2.5, 10^{-2}, 10^4)$	0.9825 $(1, 10^5, 10^{-5}, 10^4)$
ecoli-0-4-6_vs_5	0.8571 $(0.1, 10^4)$	0.8571 $(1, 10^4)$	0.8481 $(10^{-2}, 10^4)$	0.8571 $(2, 10^{-5}, 10^3)$	0.9286 $(1.5, 10^{-5}, 10^{-4}, 10^2)$
ecoli-0-6-7_vs_5	0.7418 $(10^{-2}, 10^3)$	0.8251 $(1, 10^4)$	0.7923 $(10^{-5}, 10^4)$	0.7842 $(2.5, 10^{-5}, 10^4)$	0.8005 $(1.5, 10^4, 10^{-4}, 10^4)$
ecoli4	1 $(10, 10^2)$	0.9 $(0.1, 0.1)$	1 $(1, 10^2)$	0.9588 $(0.5, 10^{-2}, 10^2)$	0.9794 $(1, 10, 10^{-3}, 10^2)$
glass5	0.75 $(1, 10)$	0.7188 $(10, 10)$	0.75 $(1, 10^2)$	0.8516 $(2.5, 10^{-3}, 10^3)$	0.9375 $(2.5, 10^5, 10^{-5}, 10^2)$
heart-stat	0.8435 $(1, 10^4)$	0.8435 $(1, 10^4)$	0.8684 $(1, 10^4)$	0.8292 $(0.5, 1, 10^4)$	0.8435 $(2, 10, 10^{-5}, 10^5)$
iono	0.9575 $(10^{-3}, 1)$	0.8931 $(0.1, 10)$	0.8938 $(10^{-2}, 10^2)$	0.9514 $(0.5, 10^{-3}, 1)$	0.9367 $(2.5, 10^{-2}, 10^{-2}, 1)$
monk2	0.7349 $(10, 10^2)$	0.7222 $(1, 10)$	0.7385 $(10^4, 10^2)$	0.6899 $(0.5, 10^{-3}, 10)$	0.7365 $(0.5, 10^2, 10^{-4}, 10)$
new-thyroid1	1 $(1, 10^3)$	0.9649 $(10^2, 10^3)$	0.9444 $(10^{-5}, 10^3)$	0.9357 $(0.5, 0.1, 10^5)$	1 $(0.5, 0.1, 10^{-4}, 10^4)$
ripley	0.9178 $(10^{-2}, 1)$	0.9079 $(10^{-5}, 0.01)$	0.923 $(0.1, 0.1)$	0.9254 $(2, 1, 0.1)$	0.9151 $(0.5, 0.1, 10^{-2}, 0.1)$
segment0	0.9892 $(10, 10^4)$	0.99 $(10^{-2}, 10^3)$	0.9842 $(10^{-2}, 10^3)$	0.9916 $(1, 0.1, 10^4)$	0.9892 $(0.5, 10^{-2}, 10^{-3}, 10^2)$
shuttle-6_vs_2-3	0.75 $(10^{-5}, 10^5)$	0.75 $(10^{-4}, 10^5)$	0.75 $(10^{-5}, 10^5)$	0.75 $(0.5, 10^{-5}, 10^5)$	1 $(0.5, 10^{-5}, 10^{-5}, 10^4)$
shuttle-c0-vs-c4	1 $(10^{-2}, 10^5)$	0.9595 $(0.1, 10^3)$	0.9865 $(1, 10^5)$	1 $(2.5, 0.1, 10^4)$	0.9865 $(2.5, 10^4, 10^{-4}, 10^4)$
sonar	0.7217 $(10, 10)$	0.7074 $(1, 10)$	0.7217 $(10, 10)$	0.733 $(0.5, 10^2, 1)$	0.7276 $(1, 1, 10^{-4}, 10)$
transfusion	0.6255 $(10, 10^5)$	0.6116 $(1, 10^3)$	0.6065 $(0.1, 10^5)$	0.6146 $(0.5, 0.1, 10^4)$	0.6113 $(0.5, 10^2, 10^2, 10^3)$
vehicle1	0.8271 $(10^2, 10^5)$	0.7053 $(1, 10^3)$	0.7588 $(1, 10^4)$	0.8284 $(0.5, 0.1, 10^4)$	0.7684 $(2, 10, 10^{-5}, 10^4)$
vehicle2	0.9974 $(1, 10^4)$	0.941 $(0.1, 10^3)$	0.9893 $(1, 10^4)$	0.9922 $(2.5, 0.1, 10^4)$	0.9974 $(0.5, 10^2, 10^{-5}, 10^4)$
vowel	0.7778 $(10^{-4}, 1)$	0.7407 $(10^{-4}, 1)$	0.7704 $(10^{-2}, 1)$	0.9389 $(0.5, 10^{-3}, 1)$	0.7778 $(0.5, 10^{-5}, 10^{-5}, 1)$
Average-Accuracy	0.8445	0.816	0.8287	0.846	0.8732
Average-Rank	2.7759	3.7759	3.2586	3	2.1897
Overall Win-Tie-Loss	2-0-4	2-0-13	4-0-6	5-0-4	11-0-0

The generalization performance of the proposed FLSPTSVM-CIL and the baseline models is evaluated on CN vs AD subjects and is given in Table-4. One can see that the average AUC of the proposed FLSPTSVM-CIL is better as compared to LSTSVM model. However, the generalization of other models is better as compared to the proposed FLSPTSVM-CIL model. As there is no free lunch theorem, the performance of the proposed FLSPTSVM-CIL model on CN vs AD is lower as compared to the rest of the models.

In Table-5, the performance evaluation of the proposed FLSPTSVM-CIL model and the baseline models is done on CN vs MCI subjects. Similar to CN vs AD, here the performance of the proposed FLSPTSVM-CIL model is better with respect to LSTSVM model.

In Table-6, the generalization performance of the proposed FLSPTSVM-CIL model and the proposed model is evaluated on MCI vs AD subjects. One can see from the Table-6 that the average AUC of the proposed FLSPTSVM-CIL model is better as compared to all the baseline models. Also, the average rank of the proposed FLSPTSVM-CIL model is lowest of all, hence, the generalization performance of the proposed FLSPTSVM-CIL model is better both in terms of average AUC and average rank.

From the above analysis, one can see that the performance of the proposed FLSPTSVM-CIL is better in MCI vs AD, which is hard to classify [52].

4.2.2. Classification of Breast cancer subjects

We evaluated the given models for classification of breast cancer subjects. We use BreakHis [53] breast cancer dataset which includes 1240 images with $400X$ magnification. The data includes two categories: benign and malignant. The benign includes subclasses: adenosis (AN), fibroadenoma (FA), phyllodes tumor (PT), and tubular adenoma (TA) having $106, 237, 115$, and 130 images, respectively. The subclasses in malignant are ductal carcinoma (DC), lobular carcinoma (LC), mucinous carcinoma (MC), papillary carcinoma (PC) with $208, 137, 169$, and 138 images, respectively. To extract the features, we followed the same pipeline as given in [54]. We evaluated the generalization performance of the proposed FLSPTSVM-CIL and the baseline models for pairwise classification of the subclasses of the benign and subclasses of malignant images. The details of the number of samples and the feature size of each pairwise subclass classification datasets is given in Table-7.

The experimental results of the classification of breast cancer subjects are given in Table-8. The accuracy of the proposed FLSPTSVM-CIL on AN vs

Table 3: Dataset specifications of the subjects of Alzheimer's disease. Here, CN: control normal, AD: Alzheimer's disease, MCI: mild cognitive impairment.

CN vs AD				
Features (%)	Train	Test	IR-All	IR-Train
5	(289×4)	(126×4)	1.21925	1.33065
10	(289×9)	(126×9)	1.21925	1.33065
15	(289×13)	(126×13)	1.21925	1.33065
20	(289×18)	(126×18)	1.21925	1.33065
25	(289×22)	(126×22)	1.21925	1.33065
30	(289×27)	(126×27)	1.21925	1.33065
35	(289×31)	(126×31)	1.21925	1.33065
40	(289×36)	(126×36)	1.21925	1.33065
45	(289×40)	(126×40)	1.21925	1.33065
50	(289×45)	(126×45)	1.21925	1.33065
100	(289×91)	(126×91)	1.21925	1.33065
CN vs MCI				
Features (%)	Train	Test	IR-All	IR-Train
5	(437×4)	(189×4)	0.572864	0.577617
10	(437×9)	(189×9)	0.572864	0.577617
15	(437×13)	(189×13)	0.572864	0.577617
20	(437×18)	(189×18)	0.572864	0.577617
25	(437×22)	(189×22)	0.572864	0.577617
30	(437×27)	(189×27)	0.572864	0.577617
35	(437×31)	(189×31)	0.572864	0.577617
40	(437×36)	(189×36)	0.572864	0.577617
45	(437×40)	(189×40)	0.572864	0.577617
50	(437×45)	(189×45)	0.572864	0.577617
100	(437×91)	(189×91)	0.572864	0.577617
MCI vs AD				
Features (%)	Train	Test	IR-All	IR-Train
5	(408×4)	(177×4)	2.12834	2.04478
10	(408×9)	(177×9)	2.12834	2.04478
15	(408×13)	(177×13)	2.12834	2.04478
20	(408×18)	(177×18)	2.12834	2.04478
25	(408×22)	(177×22)	2.12834	2.04478
30	(408×27)	(177×27)	2.12834	2.04478
35	(408×31)	(177×31)	2.12834	2.04478
40	(408×36)	(177×36)	2.12834	2.04478
45	(408×40)	(177×40)	2.12834	2.04478
50	(408×45)	(177×45)	2.12834	2.04478
100	(408×91)	(177×91)	2.12834	2.04478

Table 4: The classification performance of the models on CN vs AD subjects of Alzheimer's disease

Feaure(%)	TSVM	LSTSVM	FTWSVM	RFLSTSVM-CIL	Proposed FLSPTSVM-CIL
	AUC	AUC	AUC	AUC	AUC
	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)
5	0.7381	0.754	0.7381	0.7222	0.7222
	(0.6667, 0.8095)	(0.6349, 0.873)	(0.6349, 0.8413)	(0.6349, 0.8095)	(0.7302, 0.7143)
10	0.7619	0.7143	0.7619	0.7778	0.7222
	(0.7143, 0.8095)	(0.4444, 0.9841)	(0.6984, 0.8254)	(0.8254, 0.7302)	(0.6032, 0.8413)
15	0.8333	0.8254	0.8333	0.7937	0.8016
	(0.7619, 0.9048)	(0.8095, 0.8413)	(0.7778, 0.8889)	(0.7619, 0.8254)	(0.7619, 0.8413)
20	0.8254	0.8175	0.8254	0.8651	0.8254
	(0.8095, 0.8413)	(0.7937, 0.8413)	(0.7778, 0.873)	(0.7937, 0.9365)	(0.8095, 0.8413)
25	0.8413	0.7619	0.8413	0.8254	0.7937
	(0.7619, 0.9206)	(0.5238, 1)	(0.7778, 0.9048)	(0.8254, 0.8254)	(0.8254, 0.7619)
30	0.8492	0.7778	0.8254	0.8413	0.8016
	(0.7778, 0.9206)	(0.5556, 1)	(0.7937, 0.8571)	(0.746, 0.9365)	(0.8095, 0.7937)
35	0.8333	0.8492	0.8492	0.8175	0.7937
	(0.7937, 0.873)	(0.746, 0.9524)	(0.8095, 0.8889)	(0.8254, 0.8095)	(0.8095, 0.7778)
40	0.8571	0.8413	0.8333	0.8095	0.8254
	(0.7937, 0.9206)	(0.8095, 0.873)	(0.7937, 0.873)	(0.8095, 0.8095)	(0.8095, 0.8413)
45	0.8571	0.5873	0.8333	0.8095	0.8254
	(0.7937, 0.9206)	(0.1746, 1)	(0.7937, 0.873)	(0.8095, 0.8095)	(0.8095, 0.8413)
50	0.8571	0.5714	0.8333	0.8095	0.8254
	(0.7937, 0.9206)	(0.1429, 1)	(0.7937, 0.873)	(0.8095, 0.8095)	(0.8095, 0.8413)
100	0.8571	0.8413	0.8333	0.8095	0.8254
	(0.7937, 0.9206)	(0.8095, 0.873)	(0.7937, 0.873)	(0.8095, 0.8095)	(0.8095, 0.8413)
Average AUC	0.8283	0.7583	0.8189	0.8074	0.7965
Average Rank	1.7273	3.5909	2.3182	3.5	3.8636
Overall Win-Tie-Loss	5-0- 0	1-0- 6	0-0- 0	2-0- 3	0-0- 1

Sens. denotes sensitivity, Spec. denotes specificity

Table 5: The classification performance of the models on CN vs MCI subjects of Alzheimer's disease

Feaure(%)	TSVM	LSTSVM	FTWSVM	RFLSTSVM-CIL	Proposed FLSPTSVM-CIL
	AUC	AUC	AUC	AUC	AUC
	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)
5	0.5788	0.5413	0.5816	0.6074	0.6082
	(0.4959, 0.6618)	(0.0826, 1)	(0.4132, 0.75)	(0.3471, 0.8676)	(0.4959, 0.7206)
10	0.6087	0.4876	0.6032	0.6078	0.593
	(0.5702, 0.6471)	(0.9752, 0)	(0.4711, 0.7353)	(0.4215, 0.7941)	(0.5537, 0.6324)
15	0.6289	0.6336	0.6569	0.6606	0.6173
	(0.5372, 0.7206)	(0.3554, 0.9118)	(0.5785, 0.7353)	(0.5124, 0.8088)	(0.6612, 0.5735)
20	0.6569	0.5289	0.6431	0.6615	0.6569
	(0.5785, 0.7353)	(0.0579, 1)	(0.595, 0.6912)	(0.5289, 0.7941)	(0.5785, 0.7353)
25	0.6923	0.5	0.6794	0.6427	0.6247
	(0.6198, 0.7647)	(0, 1)	(0.6529, 0.7059)	(0.5207, 0.7647)	(0.6612, 0.5882)
30	0.656	0.5083	0.69	0.6574	0.6619
	(0.562, 0.75)	(0.0165, 1)	(0.6446, 0.7353)	(0.5207, 0.7941)	(0.6033, 0.7206)
35	0.6234	0.4867	0.6665	0.6335	0.6234
	(0.5702, 0.6765)	(0.9587, 0.0147)	(0.686, 0.6471)	(0.4876, 0.7794)	(0.5702, 0.6765)
40	0.6192	0.6441	0.6197	0.6326	0.6004
	(0.562, 0.6765)	(0.3471, 0.9412)	(0.6364, 0.6029)	(0.4711, 0.7941)	(0.5537, 0.6471)
45	0.6192	0.5957	0.6197	0.6326	0.6004
	(0.562, 0.6765)	(0.7355, 0.4559)	(0.6364, 0.6029)	(0.4711, 0.7941)	(0.5537, 0.6471)
50	0.6192	0.6422	0.6197	0.6326	0.6004
	(0.562, 0.6765)	(0.5785, 0.7059)	(0.6364, 0.6029)	(0.4711, 0.7941)	(0.5537, 0.6471)
100	0.6192	0.5957	0.6197	0.6326	0.6004
	(0.562, 0.6765)	(0.7355, 0.4559)	(0.6364, 0.6029)	(0.4711, 0.7941)	(0.5537, 0.6471)
Average AUC	0.6293	0.5604	0.6363	0.6365	0.617
Average Rank	3.0909	4.0909	2.3636	1.8182	3.6364
Overall Win-Tie-Loss	2-0- 0	2-0- 8	2-0- 0	4-0- 0	1-0- 3

Sens. denotes sensitivity, Spec. denotes specificity

Table 6: The classification performance of the models on MCI vs AD subjects of Alzheimer's disease

Feaure(%)	TSVM AUC (Sens., Spec.)	LSTSVM AUC (Sens., Spec.)	FTWSVM AUC (Sens., Spec.)	RFLSTSVM-CIL AUC (Sens., Spec.)	Proposed FLSPTSVM-CIL AUC (Sens., Spec.)
5	0.515 (0.1509, 0.879)	0.504 (1, 0.0081)	0.6432 (0.5283, 0.7581)	0.6419 (0.566, 0.7177)	0.6299 (0.6792, 0.5806)
10	0.642 (0.6226, 0.6613)	0.5444 (1, 0.0887)	0.6233 (0.7547, 0.4919)	0.6408 (0.7736, 0.5081)	0.6608 (0.6604, 0.6613)
15	0.6526 (0.5472, 0.7581)	0.6445 (0.4906, 0.7984)	0.6608 (0.6038, 0.7177)	0.6271 (0.5283, 0.7258)	0.6757 (0.6981, 0.6532)
20	0.6756 (0.6415, 0.7097)	0.5444 (1, 0.0887)	0.6163 (0.4906, 0.7419)	0.6732 (0.8302, 0.5161)	0.6636 (0.6981, 0.629)
25	0.6271 (0.5849, 0.6694)	0.5866 (0.4151, 0.7581)	0.6459 (0.566, 0.7258)	0.6597 (0.8113, 0.5081)	0.6665 (0.8491, 0.4839)
30	0.6554 (0.6415, 0.6694)	0.6228 (0.3585, 0.8871)	0.6648 (0.6038, 0.7258)	0.6732 (0.8302, 0.5161)	0.6731 (0.7736, 0.5726)
35	0.6407 (0.6604, 0.621)	0.5 (1, 0)	0.6406 (0.6038, 0.6774)	0.6678 (0.8113, 0.5242)	0.6786 (0.8491, 0.5081)
40	0.6299 (0.6226, 0.6371)	0.6755 (0.5849, 0.7661)	0.6244 (0.5472, 0.7016)	0.6597 (0.8113, 0.5081)	0.6489 (0.7736, 0.5242)
45	0.6299 (0.6226, 0.6371)	0.5823 (0.2453, 0.9194)	0.6244 (0.5472, 0.7016)	0.6597 (0.8113, 0.5081)	0.6489 (0.7736, 0.5242)
50	0.6299 (0.6226, 0.6371)	0.6203 (0.4906, 0.75)	0.6244 (0.5472, 0.7016)	0.6597 (0.8113, 0.5081)	0.6489 (0.7736, 0.5242)
100	0.6299 (0.6226, 0.6371)	0.5769 (0.2264, 0.9274)	0.6244 (0.5472, 0.7016)	0.6597 (0.8113, 0.5081)	0.6489 (0.7736, 0.5242)
Average AUC	0.6413	0.5898	0.6349	0.6581	0.6614
Average Rank	3	4.5	3.7	2	1.8
Overall Win-Tie-Loss	1-0- 0	1-0- 8	1-0- 0	4-0- 3	4-0- 2

Sens. denotes sensitivity, Spec. denotes specificity

Table 7: Dataset specifications of the subjects of Breast cancer disease. Here, AN:Adenosis, FA:Fibroadenoma, PT:Phyllodes tumor, TA:Tubular adenoma, DC:Ductal carcinoma, LC:Lobular carcinoma, MC:Mucinous carcinoma, PC: Papillary carcinoma.

Subjects	Train	Test	IR-All	IR-Train
AN vs DC	(218 × 768)	(96 × 768)	1.96226	2.15942
AN vs LC	(169 × 768)	(74 × 768)	1.29245	1.48529
AN vs MC	(191 × 768)	(84 × 768)	1.59434	1.80882
AN vs PC	(169 × 768)	(75 × 768)	1.30189	1.48529
FA vs DC	(310 × 768)	(135 × 768)	0.877637	0.91358
FA vs LC	(260 × 768)	(114 × 768)	0.578059	0.585366
FA vs MC	(283 × 768)	(123 × 768)	0.71308	0.736196
FA vs PC	(261 × 768)	(114 × 768)	0.582278	0.591463
PT vs DC	(225 × 768)	(98 × 768)	1.8087	1.96053
PT vs LC	(175 × 768)	(77 × 768)	1.1913	1.36486
PT vs MC	(197 × 768)	(87 × 768)	1.46957	1.62667
PT vs PC	(176 × 768)	(77 × 768)	1.2	1.37838
TA vs DC	(235 × 768)	(103 × 768)	1.6	1.86585
TA vs LC	(185 × 768)	(82 × 768)	1.05385	1.2561
TA vs MC	(208 × 768)	(91 × 768)	1.3	1.47619
TA vs PC	(186 × 768)	(82 × 768)	1.06154	1.26829

LC subjects achieved 59% accuracy followed by FTWSVM with 57%. In AN vs PC subjects, the proposed FLSPTSVM-CIL achieved highest 75% accuracy and the lowest by LSTSVM with 50% accuracy. Similarly, in FA vs LC subjects 70% accuracy is achieved by the proposed FLSPTSVM-CIL which is highest among the baseline models. In PT vs PC and TA vs DC, the proposed FLSPTSVM-CIL achieved the maximum accuracy with 74% and 77%, respectively, among the baseline models. One can see that the generalization performance of the proposed FLSPTSVM-CIL is better as compared to the baseline models. The proposed FLSPTSVM-CIL achieved highest average AUC. Also, the average rank of the proposed FLSPTSVM-CIL is lowest as compared to all the baseline models except FTWSVM model. In terms of overall win-tie-loss performance comparison, the proposed FLSPTSVM-CIL model achieved highest number of overall wins 5 as compared to the baseline models.

Table 8: The classification performance of the models on Breast Cancer benchmark dataset

Subjects	TSVM	LSTSVM	FTWSVM	RFLSTSVM-CIL	Proposed FLSPTSVM-CIL
	AUC	AUC	AUC	AUC	AUC
	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)	(Sens., Spec.)
AN vs DC	0.8901	0.8106	0.8511	0.822	0.8461
	(0.8649, 0.9153)	(0.7568, 0.8644)	(0.8378, 0.8644)	(1, 0.6441)	(0.8108, 0.8814)
AN vs LC	0.5607	0.5475	0.5746	0.4788	0.5906
	(0.3158, 0.8056)	(0.2895, 0.8056)	(0.3158, 0.8333)	(0.7632, 0.1944)	(0.7368, 0.4444)
AN vs MC	0.611	0.5847	0.5932	0.6413	0.6407
	(0.5263, 0.6957)	(0.4737, 0.6957)	(0.4474, 0.7391)	(1, 0.2826)	(0.8684, 0.413)
AN vs PC	0.7457	0.5	0.6799	0.6369	0.7582
	(0.8158, 0.6757)	(0, 1)	(0.6842, 0.6757)	(0.8684, 0.4054)	(0.8947, 0.6216)
FA vs DC	0.845	0.7633	0.86	0.8517	0.8133
	(0.84, 0.85)	(0.8267, 0.7)	(0.8533, 0.8667)	(0.8533, 0.85)	(0.8267, 0.8)
FA vs LC	0.6168	0.6221	0.6306	0.6099	0.7098
	(0.7945, 0.439)	(0.7808, 0.4634)	(0.6027, 0.6585)	(0.7808, 0.439)	(0.7123, 0.7073)
FA vs MC	0.5285	0.5	0.5557	0.5256	0.5454
	(0.6081, 0.449)	(1, 0)	(0.6216, 0.4898)	(0.4595, 0.5918)	(0.6622, 0.4286)
FA vs PC	0.5337	0.5673	0.5765	0.6574	0.6473
	(0.726, 0.3415)	(0.6712, 0.4634)	(0.6164, 0.5366)	(0.5342, 0.7805)	(0.6849, 0.6098)
PT vs DC	0.8977	0.8977	0.8722	0.8807	0.8807
	(0.8462, 0.9492)	(0.8462, 0.9492)	(0.8462, 0.8983)	(0.8462, 0.9153)	(0.8462, 0.9153)
PT vs LC	0.4692	0.4658	0.5024	0.56	0.5552
	(0.2439, 0.6944)	(0.2927, 0.6389)	(0.3659, 0.6389)	(0.5366, 0.5833)	(0.8049, 0.3056)
PT vs MC	0.5237	0.5686	0.596	0.621	0.5601
	(0.175, 0.8723)	(0.35, 0.7872)	(0.575, 0.617)	(0.625, 0.617)	(0.95, 0.1702)
PT vs PC	0.607	0.6839	0.6386	0.6264	0.7463
	(0.6585, 0.5556)	(0.9512, 0.4167)	(0.8049, 0.4722)	(0.7805, 0.4722)	(0.8537, 0.6389)
TA vs DC	0.7371	0.7553	0.7566	0.7553	0.7712
	(0.5833, 0.8909)	(0.5833, 0.9273)	(0.6042, 0.9091)	(0.5833, 0.9273)	(0.8333, 0.7091)
TA vs LC	0.7782	0.7114	0.7365	0.6422	0.6642
	(0.7917, 0.7647)	(0.6875, 0.7353)	(0.7083, 0.7647)	(0.6667, 0.6176)	(0.9167, 0.4118)
TA vs MC	0.4643	0.585	0.6164	0.5524	0.6155
	(0.2174, 0.7111)	(0.3478, 0.8222)	(0.5217, 0.7111)	(0.2826, 0.8222)	(0.6087, 0.6222)
TA vs PC	0.6593	0.5441	0.565	0.5607	0.5147
	(0.5833, 0.7353)	(0.5, 0.5882)	(0.5417, 0.5882)	(0.5625, 0.5588)	(0.5, 0.5294)
Average AUC	0.6543	0.6317	0.6628	0.6514	0.6787
Average Rank	3.2188	3.8125	2.375	3.1875	2.4063
Overall Win-Tie-Loss	(3-0-5)	(0-0-6)	(3-0-1)	(4-0-3)	(5-0-1)

Sens. denotes sensitivity, Spec. denotes specificity

5. Conclusion and Future work

In this paper, we formulate a novel fuzzy least squares projection twin support vector machines for class imbalance learning (FLSPTSVM-CIL) which constructs two hyperplanes such that the samples of each class are clustered around its corresponding mean and are as far as possible from the samples of other class. To handle the class imbalance problems, we incorporated fuzzy membership weights to overcome the bias towards the majority class samples. Unlike twin support vector machines which solves the dual problem by optimizing the quadratic programming problem, the proposed FLSPTSVM-CIL model obtains the two optimal hyperplanes by solving a system of linear equations. Experimental results show that the performance of the proposed FLSPTSVM-CIL model is better as compared to the baseline models. We also evaluated the performance of the proposed FLSPTSVM-CIL model on real world datasets which include classification of Alzheimer's disease patients and the classification of the breast cancer patients. Experimental results show that the generalization performance of the proposed FLSPTSVM-CIL model for the classification of the breast cancer patients and the mild cognitive impairment versus Alzheimer's data is better as compared to the baseline models. In future, we look forward to extend the proposed FLSPTSVM-CIL model to large scale problems. The codes will be available at https://github.com/mtanveer1.

Acknowledgment

This work is supported by Science and Engineering Research Board (SERB), Government of India under Ramanujan Fellowship Scheme, Grant No. SB/S2 /RJN-001/2016, and Department of Science and Technology under Interdisciplinary Cyber Physical Systems (ICPS) Scheme grant no. DST/ICPS/CPS-Individual/2018/276. We also acknowledge Council of Scientific & Industrial Research (CSIR), New Delhi, INDIA for funding under Extra Mural Research (EMR) Scheme grant no. 22(0751)/17/EMR-II. We gratefully acknowledge the Indian Institute of Technology Indore for providing facilities and support. The collection of data and sharing of this project was funded by the Alzheimer's Disease Neuroimaging Initiative (ADNI) (National Institutes of HealthGrant U01 AG024904), and DOD ADNI (Department of Defense award number W81XWH-12–2-0012). The funding for ADNIis provided by the National Institute on Aging, the National Institute of Biomedical Imaging and Bioengineering, and through generous contributions from the following: AbbVie, Alzheimer's Association; Alzheimer's Drug Discovery Foundation; Araclon Biotech; BioClinica, Inc.; Biogen; Bristol-Myers Squibb Company; CereSpir, Inc.; Cogstate; Eisai Inc.; Elan Pharmaceuticals, Inc.; Eli Lilly and Company; EuroImmun; F. Hoffmann-La Roche Ltd and its affiliated company Genentech, Inc.;Fujirebio; GE Healthcare; IXICO Ltd.; Janssen Alzheimer Immunotherapy Research & Development, LLC.; Johnson & Johnson Pharmaceutical Research & Development LLC.; Lumosity; Lundbeck; Merck & Co., Inc.; Meso Scale Diagnostics, LLC.; NeuroRx Research; Neurotrack Technologies; Novartis Pharmaceuticals Corporation; Pfizer Inc.; Piramal Imaging; Servier; Takeda Pharmaceutical Company; and Transition Therapeutics. The Canadian Institutes of Health Research is providing funds to sup-port ADNI clinical sites in Canada. Private sector contributions are facilitated by the Foundation for the National Institutes of Health (www.fnih.org). The grantee organization is the Northern California Institute for Research and Education, and thestudy is coordinated by the Alzheimer's Therapeutic Research Institute at the University of Southern California. The dissemination of ADNI data is carried out by the Laboratory for Neuro Imaging at the University of Southern California.

References

[1] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[2] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[3] Edgar Osuna, Robert Freund, and Federico Girosit. Training support vector machines: an application to face detection. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern recognition*, pages 130–136. IEEE, 1997.

[4] P Jonathon Phillips. Support vector machines applied to face recognition. In *Advances in Neural Information Processing Systems*, pages 803–809, 1999.

[5] Philipp Michel and Rana El Kaliouby. Real time facial expression recognition in video using support vector machines. In *Proceedings of the*

5th International Conference on Multimodal Interfaces, pages 258–264. ACM, 2003.

[6] Michael Schmidt and Herbert Gish. Speaker identification via support vector classifiers. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 1, pages 105–108. IEEE, 1996.

[7] Latifur Khan, Mamoun Awad, and Bhavani Thuraisingham. A new intrusion detection system using support vector machines and hierarchical clustering. *The VLDB journal*, 16(4):507–521, 2007.

[8] Mobyen Uddin Ahmed, Staffan Brickman, Alexander Dengg, Niklas Fasth, Marko Mihajlovic, and Jacob Norman. A machine learning approach to classify pedestrians' events based on IMU and GPS. *International Journal of Artificial Intelligence*, 17(2):154–167, 2019.

[9] Iman Beheshti, M. A. Ganaie, Vardhan Paliwal, Aryan Rastogi, Imran Razzak, and M. Tanveer. Predicting brain age using machine learning algorithms: A comprehensive evaluation. *IEEE Journal of Biomedical and Health Informatics*, 10.1109/JBHI.2021.3083187.

[10] Olvi L Mangasarian and Edward W Wild. Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(1):69–74, 2005.

[11] Jayadeva, R. Khemchandani, and S. Chandra. Twin support vector machines for pattern classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):905–910, 2007.

[12] M. Tanveer. Robust and sparse linear programming twin support vector machines. *Cognitive Computation*, 7(1):137–149, 2015.

[13] M. Tanveer. Application of smoothing techniques for linear programming twin support vector machines. *Knowledge and Information Systems*, 45(1):191–214, 2015.

[14] M. A. Ganaie and M. Tanveer. Robust general twin support vector machine with pinball loss function. In *Machine Learning for Intelligent Multimedia Analytics*, pages 103–125. Springer, 2021.

[15] M. Tanveer, T. Rajani, and M. A. Ganaie. Improved sparse pinball twin SVM. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 3287–3291. IEEE, 2019.

[16] Manisha Singla, Debdas Ghosh, KK Shukla, and Witold Pedrycz. Robust twin support vector regression based on rescaled hinge loss. *Pattern Recognition*, 105:107395, 2020.

[17] M. Tanveer, K. Shubham, Mujahed Aldhaifallah, and Shen Shyang Ho. An efficient regularized K-nearest neighbor based weighted twin support vector regression. *Knowledge-Based Systems*, 94:70–87, 2016.

[18] M. Tanveer and K Shubham. A regularization on Lagrangian twin support vector regression. *International Journal of Machine Learning and Cybernetics*, 8(3):807–821, 2017.

[19] S. Balasundaram and M. Tanveer. On Lagrangian twin support vector regression. *Neural Computing and Applications*, 22(1):257–267, 2013.

[20] M. A. Ganaie, M. Tanveer, and P. N. Suganthan. Oblique decision tree ensemble via twin bounded SVM. *Expert Systems with Applications*, 143:113072, 2020.

[21] M. A. Ganaie and M. Tanveer. LSTSVM classifier with enhanced features from pre-trained functional link network. *Applied Soft Computing*, 93:106305, 2020.

[22] M. Tanveer, M. A. Ganaie, and P. N. Suganthan. Ensemble of classification models with weighted functional link network. *Applied Soft Computing*, 107:107322, 2021.

[23] M. Tanveer, A. Sharma, and P. N. Suganthan. Least squares KNN-based weighted multiclass twin SVM. *Neurocomputing*, https://doi.org/10.1016/j.neucom.2020.02.132.

[24] Chun-Fu Lin and Sheng-De Wang. Fuzzy support vector machines. *IEEE Transactions on Neural Networks*, 13(2):464–471, 2002.

[25] Daisuke Tsujinishi and Shigeo Abe. Fuzzy least squares support vector machines for multiclass problems. *Neural Networks*, 16(5-6):785–792, 2003.

[26] Yongqiao Wang, Shouyang Wang, and Kin Keung Lai. A new fuzzy support vector machine to evaluate credit risk. *IEEE Transactions on Fuzzy Systems*, 13(6):820–831, 2005.

[27] S. Balasundaram and M. Tanveer. On proximal bilateral-weighted fuzzy support vector machine classifiers. *International Journal of Advanced Intelligence Paradigms*, 4(3-4):199–210, 2012.

[28] Zhenning Wu, Huaguang Zhang, and Jinhai Liu. A fuzzy support vector machine algorithm for classification based on a novel PIM fuzzy clustering method. *Neurocomputing*, 125:119–124, 2014.

[29] Wenjuan An and Mangui Liang. Fuzzy support vector machine based on within-class scatter for classification problems with outliers or noises. *Neurocomputing*, 110:101–110, 2013.

[30] B. Richhariya and M. Tanveer. A reduced universum twin support vector machine for class imbalance learning. *Pattern Recognition*, 102:107150, 2020.

[31] Radu-Emil Precup, Teodor-Adrian Teban, Adriana Albu, Alexandra-Bianca Borlea, Iuliu Alexandru Zamfirache, and Emil M Petriu. Evolving fuzzy models for prosthetic hand myoelectric-based control. *IEEE Transactions on Instrumentation and Measurement*, 69(7):4625–4636, 2020.

[32] Benjamin X Wang and Nathalie Japkowicz. Boosting support vector machines for imbalanced data sets. *Knowledge and Information Systems*, 25(1):1–20, 2010.

[33] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[34] B. Richhariya and M. Tanveer. A robust fuzzy least squares twin support vector machine for class imbalance learning. *Applied Soft Computing*, 71:418–432, 2018.

[35] M. A. Ganaie, M. Tanveer, and P. N. Suganthan. Regularized robust fuzzy least squares twin support vector machine for class imbalance

learning. In *2020 International Joint Conference on Neural Networks, IJCNN*, pages 1–8. IEEE, 2020.

[36] M. Tanveer, A. Sharma, and P. N. Suganthan. General twin support vector machine with pinball loss function. *Information Sciences*, 494:311–327, 2019.

[37] Qiaolin Ye, Chunxia Zhao, Ning Ye, and Yannan Chen. Multi-weight vector projection support vector machines. *Pattern Recognition Letters*, 31(13):2006–2011, 2010.

[38] Xiaobo Chen, Jian Yang, Qiaolin Ye, and Jun Liang. Recursive projection twin support vector machine via within-class variance minimization. *Pattern Recognition*, 44(10-11):2643–2655, 2011.

[39] Yuan-Hai Shao, Nai-Yang Deng, and Zhi-Min Yang. Least squares recursive projection twin support vector machine for classification. *Pattern Recognition*, 45(6):2299–2307, 2012.

[40] Shifei Ding and Xiaopeng Hua. Recursive least squares projection twin support vector machines for nonlinear classification. *Neurocomputing*, 130:3–9, 2014.

[41] Zhi-Min Yang, He-Ji Wu, Chun-Na Li, and Yuan-Hai Shao. Least squares recursive projection twin support vector machine for multi-class classification. *International Journal of Machine Learning and Cybernetics*, 7(3):411–426, 2016.

[42] M. Tanveer, T. Rajani, R. Rastogi, and Y. H. Shao. Comprehensive review on twin support vector machines. *arXiv preprint arXiv:2105.00336*, 2021.

[43] B. Richhariya, M. Tanveer, and Alzheimer's Disease Neuroimaging Initiative. Least squares projection twin support vector clustering (LSPTSVC). *Information Sciences*, 533:1–23, 2020.

[44] Ioan-Daniel Borlea, Radu-Emil Precup, Alexandra-Bianca Borlea, and Daniel Iercan. A unified form of fuzzy C-means and K-means algorithms and its partitional implementation. *Knowledge-Based Systems*, 214:106731, 2021.

[45] Rukshan Batuwita and Vasile Palade. FSVM-CIL: fuzzy support vector machines for class imbalance learning. *IEEE Transactions on Fuzzy Systems*, 18(3):558–571, 2010.

[46] Kai Li and Hongyan Ma. A fuzzy twin support vector machine algorithm. *International Journal of Application or Innovation in Engineering and Management (IJAIEM)*, 2(3):459–465, 2013.

[47] Bin-Bin Gao, Jian-Jun Wang, Yao Wang, and Chan-Yun Yang. Coordinate descent fuzzy twin support vector machine for classification. In *2015 IEEE 14th international conference on machine learning and applications (ICMLA)*, pages 7–12. IEEE, 2015.

[48] M Arun Kumar and Madan Gopal. Least squares twin support vector machines for pattern classification. *Expert Systems with Applications*, 36(4):7535–7543, 2009.

[49] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[50] Jesús Alcalá-Fdez, Alberto Fernández, Julián Luengo, Joaquín Derrac, Salvador García, Luciano Sánchez, and Francisco Herrera. Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic & Soft Computing*, 17, 2011.

[51] B. Richhariya, M. Tanveer, A. H. Rashid, and Alzheimer's Disease Neuroimaging Initiative. Diagnosis of Alzheimer's disease using universum support vector machine based recursive feature elimination (USVM-RFE). *Biomedical Signal Processing and Control*, 59:101903, 2020.

[52] M. Tanveer, B. Richhariya, R. U. Khan, A. H. Rashid, P. Khanna, M. Prasad, and C. T. Lin. Machine learning techniques for the diagnosis of Alzheimer's disease: A review. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16(1s):1–35, 2020.

[53] Fabio A Spanhol, Luiz S Oliveira, Caroline Petitjean, and Laurent Heutte. A dataset for breast cancer histopathological image classification. *IEEE Transactions on Biomedical Engineering*, 63(7):1455–1462, 2015.

[54] Chandan Gautam, Pratik K Mishra, Aruna Tiwari, Bharat Richhariya, Hari Mohan Pandey, Shuihua Wang, M. Tanveer, and Alzheimer's Disease Neuroimaging Initiative. Minimum variance-embedded deep kernel regularized least squares method for one-class classification and its applications to biomedical data. *Neural Networks*, 123:191–216, 2020.

Highlights

1. A novel fuzzy least squares projection twin SVM is proposed for class imbalance learning.
2. The proposed algorithm implements the structural risk minimization principle.
3. The matrices appear in the proposed primal formulation are positive definite.
4. The proposed model seeks projections such that the samples of each class are clustered around its corresponding mean and the samples of different classes are as far as possible.
5. Fuzzy weights are assigned in the proposed model to handle the class imbalance problems.
6. Applications of proposed models are shown on the classification of breast cancer and Alzheimer's disease subjects.

Credit Author Statement

M.A. Ganaie: Conceptualization, Methodology, Formal Analysis, Investigation, Resources, Writing Original draft, Writing Review & editing, Visualization

M. Tanveer: Conceptualization, Methodology, Validation, Writing–Review & editing, Supervision, Funding acquisition

Declaration of interests

☐The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

The authors of this work have no competing interests.