Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2022) 30: 1337 – 1353
© TÜBİTAK
doi:10.55730/1300-0632.3852

Learning target class eigen subspace (LTC-ES) via eigen knowledge grid

Sanjay Kumar Sonbhadra*, **Sonali Agarwal**, **P. Nagabhushan**
Indian Institute of Information Technology Allahabad, Prayagraj U.P.

Received: 25.08.2021 • **Accepted/Published Online:** 27.02.2022 • **Final Version:** 31.05.2022

Abstract: In one-class classification (OCC) tasks, only the target class (class-of-interest (CoI)) samples are well defined during training, whereas the other class samples are totally absent. In OCC algorithms, the high dimensional data adds computational overhead apart from its intrinsic property of curse of dimensionality. For target class learning, conventional dimensionality reduction (DR) techniques are not suitable due to negligence of the unique statistical properties of CoI samples. In this context, the present research proposes a novel target class guided DR technique to extract the eigen knowledge grid that contains the most promising eigenvectors of variance-covariance matrix of CoI samples. In this process the lower and higher eigenvalued eigenvectors are rejected via statistical analysis because the high variance may split the target class itself, whereas the lower variance do not contribute significant information. Furthermore, the identified eigen knowledge grid is utilized to transform high dimensional samples to the lower dimensional eigen subspace. The proposed approach is named as learning target class eigen subspace (LTS-ES) that ensures strong separation of the target class from other classes. To show the effectiveness of transformed lower dimensional eigen subspace, one-class support vector machine (OCSVM) has been experimented on wide variety of benchmark datasets in presence of: original feature space, transformed features obtained via eigenvectors of approximately 80%–90% cumulative variance, transformed features obtained via knowledge grid and transformed features obtained via eigenvectors of approximately 50% cumulative variance. Finally, a new performance measure parameter called stability factor is introduced to validate the robustness of the proposed approach.

Key words: One-class classification, target class, dimensionality reduction, class-of-interest, eigen knowledge grid, one-class support vector machine

Nomenclature:

Symbol	Meaning	Symbol	Meaning	Symbol	Meaning
T_c	Target class	X	Input samples	L	Lagrangian
R	Radius of hypersphere	a	Center of hypersphere	x	Sample
w	Weight vector of OCSVM	ρ	Margin parameter	A	Associativity
var	Variance	Cov	Covariance matrix	V	Eigenvector
λ	Eigenvalue	K	Knowledge matrix	SF	Stability factor

1. Introduction

The nature of one-class classification (OCC) task is very different from conventional binary/multiclass classification problem due to presence of only the target class samples during training as shown in Figure 1. This problem was initially identified and defined as single-class classification by Minter [1], where the presence of

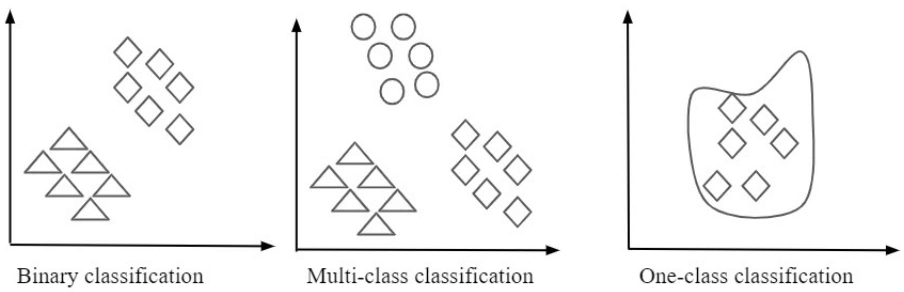
*Correspondence: rsi2017502@iiita.ac.in

This work is licensed under a Creative Commons Attribution 4.0 International License.

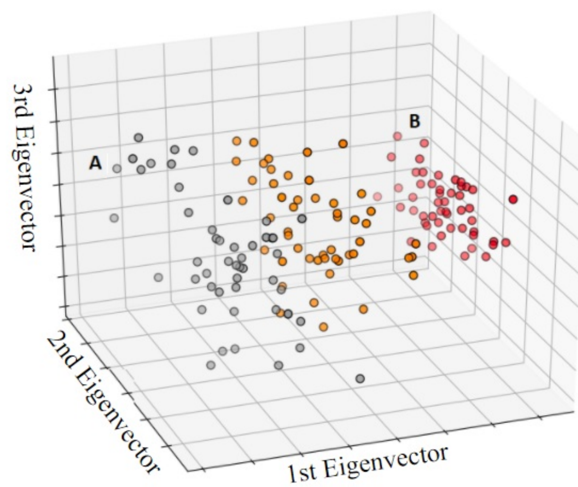
only the positive class samples makes the classification task very difficult. Later, different terminologies were used for the same problem like one-class classification (OCC), outlier detection, novelty detection and concept learning [2]. During past three decades several OCC algorithms have been proposed such as isolation forest (IF), support vector data description (SVDD), one-class nearest neighbour (OCNN), one-class support vector machine (OCSVM), etc. [3, 4]. In recent years, massive data is being generated by geographically distributed heterogeneous sources, and the performance of classifiers is influenced by the quality of input. Apart from massive sample space, the high dimensionality of the data adds computation overhead apart from its intrinsic property of curse of dimensionality [5]. In this context several training samples and dimensionality reduction algorithms have been proposed for binary/multi-class classification tasks, whereas for OCC problems, limited number of publications have been reported till date [2, 6]. In this context, the present research proposes a novel target class guided dimensionality reduction algorithm using eigenspace analysis to ensure strong separation of target class samples from outliers.

The high dimensional data suffers from curse of dimensionality [5], where the growth in number of features increases the sparsity of the samples; that may raise severe complexities while processing for some statistical significance. Due to sparsity, the similar objects seem from different classes (shown in Figure 1, where points A and B belong to same class, but seem from different classes), whereas massive computation is needed in the presence of enormous features. Moreover, the high-dimensional datasets are difficult to analyze and visualize due to noise and redundant features [7]. To overcome these issues, the high dimensional dataset must be transformed to lower dimensional space with minimum information loss.

The DR techniques project the samples from high dimension n to lower dimension p, where $p << n$. The objective is to maximize the learning ability while ensuring the reduced computation cost and enhanced visualization. The DR techniques can be categorized in two types: feature selection and feature transformation [8, 9]. In feature selection, a subset of most discriminant independent features are retained and rest are discarded via filter or wrapper approaches [10]. The filter methods do not involve model evaluation or learning, and outputs a subset of features based on certain criterion or rank [11], whereas the wrapper approaches generate a near-optimal feature subset guided by model evaluation and learning [12]. The feature transformation techniques project the samples from original feature space to another space via means of feature scaling, eigenspace analysis, etc. In this context, several eigen decomposition based DR algorithms have been offered during past decades [13, 14]. Among all DR techniques, principal component analysis (PCA) and its variations are very popular, where it is claimed that principal components (PCs) with high variance are sufficient to exhibit the behaviour of the data, but it is evident that this approach suffers from overfitting [15]. Whereas, the lower eigenvalued PCs do not contain significant information, but it is also observed that these PCs are equally important for anomaly detection tasks [16]. In presence of only CoI samples, all existing PC selection strategies may lead to following issues: higher variance may split target class itself and low variance do not guarantee the cohesiveness. With this notion, the present research proposes a novel target class guided feature transformation technique, where a novel way of rejecting the higher and lower eigenvalued eigenvectors of variance-covariance matrix of target class is offered. Furthermore, the remaining eigenvectors are treated as knowledge grid for the target class, which is further used for feature transformation. The whole process is guided by only the target class samples; hence, ensures the tighter description of CoI. In present research, OCSVM is utilized as classifier to validate the performance of proposed feature transformation method. Following are the contributions of the present research:



(a) Types of classification.



(b) Visibility in higher dimension.

Figure 1. Classification types and higher dimensional visualization (3D visualization of Iris dataset).

- Target class guided feature transformation technique is proposed via eigenspace analysis.
- A novel tweak is established to eliminate irrelevant higher and lower eigenvalued eigenvectors of variance-covariance matrix concerning the target class samples.
- The retained eigenvectors of variance-covariance matrix are treated as knowledge grid to transform the high dimensional samples to lower dimensional eigen subspace.
- The effectiveness of the proposed dimensionality reduction approach is validated over 9 benchmark datasets using OCSVM as classifier.
- A new performance measure parameter called stability factor is introduced to verify the robustness of the proposed approach.

The rest of the paper is organized as follows: Section 2 briefs about the evaluation of PCA and eigenspace analysis followed by developments in target class guided dimensionality reduction techniques and one-class support vector classifiers (OCSVCs). Section 3 describes the proposed approach. Experimental setup and results are discussed in Section 4 and Section 5 concludes the present research with possible future scope.

2. Related work

In recent years, several DR techniques have been offered for binary/multiclass classification tasks [13, 14], and it is observed that eigen decomposition is the heart of all techniques. It is also evident that towards OCC tasks, limited number of research works have been reported till date [8]. In this context, the present research proposes a target class guided DR technique using eigenspace analysis; therefore, in this section, recent developments in eigenspace analysis is discussed followed by the recent achievements in dimensionality reduction techniques towards target class mining. With intense literature survey, it is evident that one-class support vector classifiers (SVDD and OCSVM) are more effective in one-class classification tasks; hence, a brief overview of OCSVCs is also included in Section 2.2.2. Both the variants of OCSVCs (SVDD and OCSVM) are equivalent in unit norm space [17]; therefore, in the present research, OCSVM is chosen as a classifier for experiments.

2.1. Eigenspace analysis

For a dataset $X = \{x_1, x_2 \ldots x_t\}$ (where t is number of samples), PCA and its variants give the p orthonormal axes retaining maximum variance under projection. Meanwhile, several algorithms have been offered for selection of PCs to attain optimal classification performance [5, 9, 16, 18–22], but the identification of best-suited PCs is still unclear in several cases [19, 20, 23]. It is also evident that the wrong selection of PCs leads to information loss (underestimation) or noise (overestimation) [23]. Conventionally, the PCs with high variance are preferred, but it is evident that this may cause from overfitting [15]. Whereas, the lower eigenvalued PCs do not contain significant information, but it is also evident that these PCs are equally important and can be utilized for anomaly detection tasks [16]. The DR techniques are usually utilized in the presence of multiclass samples and not yet significantly explored for target class learning. In the presence of multiclass information, the existing DR techniques cannot be used to extract target class information due to influence of other class samples. In this context, the present research proposes a target class supervised feature transformation method, where a novel tweak is offered to exclude least important and misleading eigenvectors of variance-covariance matrix of the target class samples. In this approach, higher and lower eigenvalued eigenvectors are rejected using statistical analysis considering the facts that the high variance may split the target class itself whereas the lower eigenvalued vectors do not carry significant information; therefore, unable to maximize the cohesiveness. The detailed discussion on method of selection of eigenvectors to form knowledge grid guided by the target class is covered in Section 3, which is the core contribution of this research.

2.2. Towards target class mining

2.2.1. Target class guided DR techniques

Several DR techniques have been offered for conventional classification models, whereas concerning to the OCC tasks very few research articles have been reported till date [2]. Tax et al. [24] considered the distribution of outliers and target class samples as uniform and Gaussian respectively, and proved that for OCC problems, the low variance PCs are more informative. In this article, behaviour of SVDD on Gaussian distribution is analyzed and afterwards, the effect of applying PCA was discussed with extensive experiments on face [25] and concordia [26] datasets. This research exhibited that for huge sample space, lower eigenvalued PCs attained smaller error. Furthermore, Lian [27] demonstrated the importance of low-variance PCs for OCC task. Later, an approach for anomaly detection technique in video surveillance system was proposed by Liu et al. [28] using motion directional PCA, where PCA was applied to every separate directions independently and every PCA packed the motion vector features in the same direction into a lower dimensional feature vector.

In fault diagnosis of reciprocating compressors of smart healthcare systems, indicator diagram plays a crucial role. A novel approach for indicator recognition and anomaly detection was proposed by Feng et al. [28], where the discrete two-dimensional curvelet transform was used to convert an indicator diagram into feature vector. Afterwards, the nonlinear PCA was applied to map these feature vectors to 3-dimensional space. The multiclass SVM and OCSVM were chosen as the classifiers and novelty detector, respectively. The proposed approach outperformed the traditional wavelet-based approach.

Afterwards, Jeong et al. [29] proposed two feature selection (FS) approaches: SVDD-radius-recursive feature elimination (RFE) and SVDD-dual-objective-RFE. The first method utilized the square of the hypersphere's radius as a criteria function for FS and reduced the boundary size by iterative elimination of an individual feature in criteria function. Whereas, the second method maximizes the dual function of SVDD as criteria, i.e. the Lagrange's multipliers. It is observed from experiments that both of these methods exhibited similar performance. Later, Nagabhushan and Meenakshi [30] proposed a three-tier feature subsetting method guided by the target class samples. In this approach, initially less important features were removed and furthermore, redundant features were discarded and in the last stage optimum feature set was obtained from the subsequent subsets of features. This approach eliminates irrelevant features to minimize the intra-class variance and ensures tighter description of the target class.

2.2.2. One-class classification

OCC algorithms are found efficient for anomaly/novelty detection tasks where the target class is well defined and the other class samples are totally/partially absent [2, 31]. In this context OCSVCs (SVDD and OCSVM) are proven more robust in different application areas such as healthcare, document analysis, surveillance system, etc., because OCSVCs can work well in absence of negative class (outlier) samples. Therefore, OCSVM is utilized as a classifier in present research. This section briefs the variants of OCSVCs: SVDD and OCSVM.

Tax et al. [32] proposed a novel OCSVC called SVDD, where a hypersphere encloses all target class training samples. The boundary points of hypersphere are called support vectors. If a test sample falls outside of the hypersphere, SVDD treats it as an outlier and rejects (Figure 2.2.2). The SVDD is defined as follows:

$$L(R, a, \alpha_i, \gamma_i, \xi_i) = R^2 + C\sum_i \xi_i - \sum_i \alpha_i\{R^2 + \xi_i - (\parallel x_i \parallel^2 - 2a.x_i + \parallel a \parallel^2)\} - \sum_i \gamma_i \xi_i \tag{1}$$

subject to: $\parallel x_i - a \parallel^2 \leq R^2 + \xi_i$,where $\xi_i \geq 0 \ \forall$ i

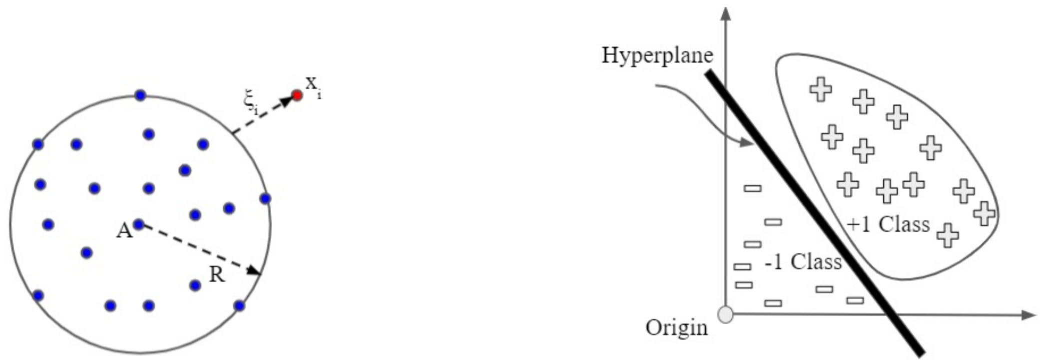
where a and R are center and radius of the hypersphere respectively. x_i is an outlier, ξ is slack variable that penalizes x_i, C controls the trade-off between the errors and volume and $\alpha_i \geq 0$, $\gamma_i \geq 0$ are Lagrange multipliers. The objective is to minimize the volume of hypersphere to enclose all target class training samples. After calculating the partial derivatives and substitution into Eq. 1, following is obtained:

$$L = \sum_i \alpha_i(x_i, x_i) - \sum_{i,j} \alpha_i \alpha_j (x_i, x_j). \tag{2}$$

If the description value of a test sample is greater than C then it is treated as an outlier. Kernels can be used to reformulate the SVDD and the output can be calculated as follows:

$$f(x) = R^2 - \|\phi(x) - a\|^2. \tag{3}$$

The output of Eq. 3 is positive for samples inside the boundary, negative for outliers and zero for boundary points.



(a) Support vector data description (SVDD). (b) One-class support vector machine (OCSVM).

Figure 2. Working principal of SVDD and OCSVM.

Later, an alternate OCC approach called one-class support vector machine (OCSVM) was offered by Schlökopf et al. [33]. In this approach, it is assumed that all negative class samples reside on the subspace of the origin and a hyperplane separates the target class with the maximal margin from the origin as shown in Figure 2.2.2. The objective function of OCSVM is defined as follows:

$$\max_{w,\xi,\rho} \frac{1}{2} \parallel w \parallel^2 + \frac{1}{\upsilon N} \sum_{i}^{N} \xi_i - \rho \tag{4}$$

subject to: $w.\phi(x_i) \geq \rho - \xi_i$ and $\xi_i \geq 0 \quad \forall \ i \in \{1, 2, \dots n\}$,
where sample x_i is represented by ϕ in feature space and outliers are penalized by the slack variable ξ_i. The hyperplane is characterized by weight vector w and margin parameter ρ, where the lower bound on the number of support vectors and upper bound on the fraction of outliers are decided by $\upsilon \ \epsilon$ (0,1]. Following is the dual optimization problem of Eq. 4:

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j K(x_i, x_j) \tag{5}$$

subject to: $0 \leq \alpha_i \leq \frac{1}{\upsilon N}$, $\sum_{i=1}^{N} \alpha_i = 1$.
where $\alpha = [\alpha_1, \alpha_2, \dots \alpha_N]^T$ and α_i is the Lagrange multiplier, whereas the weight-vector w can be expressed as:

$$w = \sum_{i=0}^{N} \alpha_i \phi(x_i). \tag{6}$$

The margin parameter ρ is computed by any x_i whose corresponding Lagrange multiplier satisfies $0 < \alpha_i < \frac{1}{\upsilon N}$

$$\rho = \sum_{j=1}^{N} \alpha_j K(x_j, x_i). \tag{7}$$

With kernel expansion the decision function can be defined as follows:

$$f(x) = \sum_{i=1}^{N} \alpha_i K(x_i, x) - \rho. \tag{8}$$

Finally, the test instance x can be labelled as follows:

$$\hat{y} = sign(f(x)), \tag{9}$$

where $sign(.)$ is sign function.

It is observed that both the SVDD and OCSVM perform equally with Gaussian kernel. In unit norm feature space, the margin of a hyperplane of OCSVM is equal to the norm of the centre of SVDD [17].

3. Proposed work

The present research offers a novel way of eigen subspace representation of the CoI samples when only the target class samples are available during training. Furthermore, OCSVM is used as classification algorithm because it has the capability to work in presence of only the target class samples [2, 6, 34]. Figure 3 shows the schematic of the overall process.

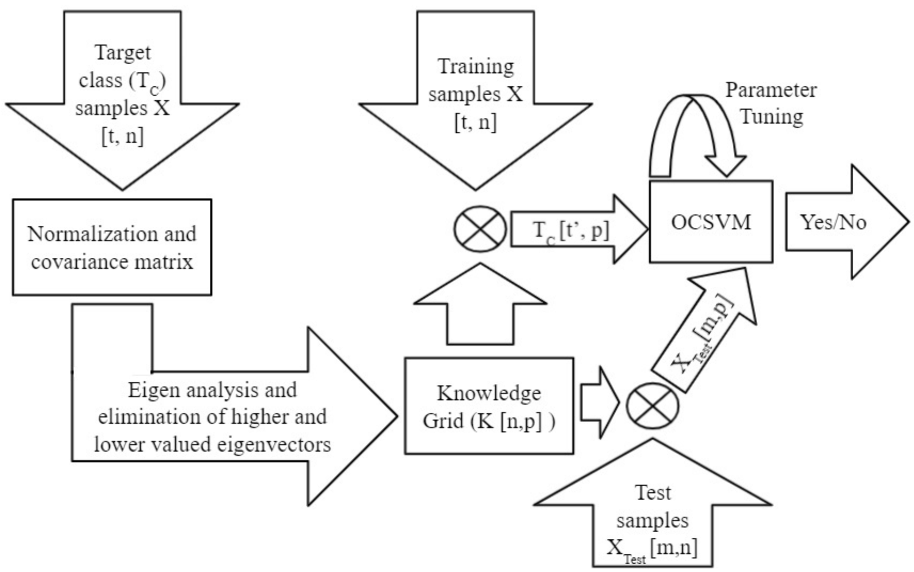


Figure 3. Schematic of proposed model for feature transformation and classification.

Initially, the eigenspace analysis is performed over the target class samples (let T_c is the target class) to compute the transformed feature subspace. The aim is to find a function F to map the target class as $Y_{EigenSpace} \leftarrow F(T_C)$, that represents the transformed form of the target class in lower dimension. The transformed features produced by F satisfy following two primary objectives: a) enhance the associativity among CoI samples and b) ensure strong separation of target class from other class samples. Statistically, these objectives are contradictory to each other because associativity (A_{CoI}) measures the proximity of the CoI samples from its mean and the associativity within the intended class is reciprocal to the variance (a measure of the scattering of the data) i.e. $A_{CoI}\ \alpha\ \frac{1}{var(CoI)}$. On the other hand, $Significance(f_{discrim})\ \alpha\ var(CoI)$. The features satisfying the above conditions also persuade the following objectives: a) maximize the target class density to avoid the false rejection of the target class, and b) minimize the false acceptance into the target class during the classification process.

Let the dataset is $X = [x_1, x_2, \cdot x_t]^T$ where t is number of samples, $x_i = \{x_i^1, x_i^2, \cdot x_i^n\}$ and classes are $Y = C_1, C_2, \cdot C_m$. Initially, target class ($T_c \in Y$) samples are selected as training set, and after normalization, variance-covariance matrix [13] is calculated using following equation:

$$Cov(T_c) = \frac{1}{t} T_C^T T_C. \tag{10}$$

The eigen decomposition is performed on variance-covariance matrix, and eigenvalues are arranged in ascending order as $\lambda_1, \lambda_2, \ldots, \lambda_n$, where the associated eigenvectors are $V_1, V_2, \ldots, V_n$ (as shown in Eq. 11). Let the range of variance (λ) is $[\underline{\lambda}, \overline{\lambda}]$.

$$V([T_c]) = [V_1, V_2, \ldots, V_n] \tag{11}$$

Algorithm 1: Learning target class eigen subspace (LTC-ES).

Input: Target class samples $T_c[t, n]$
Output:Knowledge grid (K) representing the target class Step 1: Compute mean and covariance matrix:
1.1 Mean: $\overline{T_c} = \frac{\sum_{i=0}^{t} x_{i,j}}{t}$ where $j \in 1, 2, \ldots, t$;
1.2 Variance-covariance matrix:
$COV(T_c) = \frac{1}{t} \sum (T_c - \overline{T_c})^T (T_c - \overline{T_c})$
Step 2: Eigen decomposition:
Eigenvalues: $\{\lambda_i\}$, where $i \in \{1, 2, \ldots, n\}$
Corresponding eigenvectors: $V = \{V_i\}$, where $i \in \{1, 2, \ldots, n\}$
Step 3: Selection of eigenvectors
3.1 Cumulative variance of eigenvectors: $V_{cum}^{var} = [V_1^{var}, V_2^{var}, \ldots, V_n^{var}]$
3.2 Reject higher valued eigenvectors: $V_{reject}^{high} \xleftarrow{k} \sum_{i=0}^{k} V_i^{var} \geq \sum_{i=k}^{n} V_i^{var}$
3.3 Reject lower valued eigenvectors: $V_{reject}^{low} \xleftarrow{n-(m+k)} \sum_{j=m+1}^{n} V_j^{var} \leq \sum_{i=k+1}^{m} V_i^{var}$
Step 4: Knowledge grid $K = [V_{k+1}, V_{k+2}, \ldots, V_{m-1}]$
Step 5: Transformation of input to lower dimension: $X \times K$
Step 6: End.

The promising eigenvectors are further selected based on the cumulative captured variance, where the higher and lower eigenvalued eigenvectors are rejected using Eqs. 13 and 14 because the high variance may mislead the model in presence of deceptive similar classes and low variance vectors do not carry significant information. To ensure the tightness of target class, initially, the cumulative variance is calculated using Eq 12 and the high eigenvalued vectors are rejected using Eq. 13.

$$V_{cum}^{var} = [V_1^{var}, V_2^{var}, \ldots V_n^{var}] \tag{12}$$

where V_i^{var} shows the cumulative variance of i components.

$$V_{reject}^{high} \xleftarrow{k} \sum_{i=0}^{k} V_i^{var} \geq \sum_{i=k}^{n} V_i^{var} \tag{13}$$

where k is the number of high eigenvalued eigenvectors eligible for rejection. The V^{high}_{reject} are rejected and from remaining eigenvectors, cumulative variance is calculated using Eq. 12 to eliminate lower valued eigenvectors with following equation:

$$V^{low}_{reject} \overset{n-(m+k)}{\longleftarrow} \sum_{j=m+1}^{n} V_j^{var} \leq \sum_{i=k+1}^{m} V_i^{var} \tag{14}$$

where m eigenvectors are eligible for rejection and V^{low}_{reject} are rejected. Furthermore, the remaining eigenvectors are used as knowledge grid K of the target class (T_c).

$$K[T_c] = [V_{k+1}, V_{k+2}, \ldots, V_{m-1}], \tag{15}$$

where $var(V_k) > var(V_{k+1})$. After evaluating the knowledge grid, the target samples are transformed using Eq. 16. This process transforms the target class samples from n dimension to lower dimensional eigen subspace.

$$Samples_{training} = X \times K \tag{16}$$

These transformed training samples are used to train the OCSVM. Algorithm 1 gives the description of the overall process of LTS-ES. Later, the same eigen knowledge grid K is used to transform the test samples for experiments.

4. Experiments and results

To validate the effectiveness of the offered feature transformation approach, intensive experiments are performed with 9 benchmark datasets of varying characteristics (Table 1 shows description of datasets). The experiments have been performed in the following two phases: a) feature transformation and b) classification using OCSVM. For a given target class T_c of a dataset; initially, after normalization and computation of variance-covariance matrix, eigen decomposition is performed, and the knowledge grid K is calculated to transform the feature space of training and test samples. Later, classification is performed using OCSVM in presence of: a) original features, b) transformed features obtained by higher eigenvalued eigenvectors with cumulative variance of approximately 80%–90%, c) transformed features obtained by proposed method, and d) transformed features obtained by higher valued eigenvectors with cumulative variance of approximately 50%.

Table 1. Details of utilized datasets.

S. no.	Dataset	Instances	Features	Classes
1	Iris	150	4	3
2	svmGuide1	7089	4	2
3	Credit Fraud	284807	30	2
4	Diabetes	768	9	2
5	Glass	214	10	6
6	Heart	270	13	2
7	Wine	178	13	3
8	MNIST	70,000	785	10
9	Indian Pine	10249	200	16

$$Accuracy = \frac{TN + TP}{TN + FN + TP + FP},\ Precision = \frac{TP}{TP + FP}$$
$$Specificity = \frac{TN}{TN + FP},\ Sensitivity = \frac{TP}{TP + FN} \tag{17}$$

Initially, the OCSVM is trained on 80% of the transformed target class samples, whereas the testing is done with remaining 20% in-class and other out-class samples. RBF kernel is used in experiments because it is more suitable for high dimensional datasets that exist nearby manifolds (e.g., Indian Pines) [35]. Four benchmark performance measure parameters: accuracy, precision, specificity and sensitivity are considered for evaluation of proposed approach as shown in Eq. 17. Table 2 shows the complete details of the experiments using proposed method, whereas Table 3 shows the overall comparative results using above mentioned four scenarios of selected/transformed features. Area under the receiver operating characteristics (AUROC) is used for performance evaluation with original and reduced feature space. Every class associated with each dataset are selected as target class separately, and respective AUROC score is computed. For simplicity, in Figure 4 AUROC few classes are shown for MNIST and Indian Pine datasets. From Figure 4, it is also observed that transformed feature subspace obtained by proposed method performs better for all the cases. It is also evident that for high dimensional datasets like MNIST and Indian Pine, the proposed approach performed better compare to the original features that proves its robustness. It is also evident that the proposed approach reduces the number of features to approximately 7% of the original feature space. Presence of deceptive similar classes (classes seem similar but actually they are different) makes the classification task challenging due to increased chances of false-positive. For example in MNIST dataset the digits: '0', '3', '8' and '9' sometimes appear same due to writing style, and experimental results prove that the proposed method handles such issues efficiently compared to higher eigenvalued vectors and original features (refer Table 3, Figures 4 and 5). From Table 3, it is clear that sensitivity and specificity values of proposed method is better than other feature selection methods that ensures the workability of the proposed method in presence of deceptive similar classes.

It is observed from Table 3 that the reduced features obtained by the proposed approach outperforms other ways of feature selection. It is also evident that compared to other scenarios the proposed method gains increase in average accuracy by 1.87%–4.27%, precision by 2.90%–3.09%, specificity by 2.62%–2.82% and sensitivity by 2.88%–6.76% with approximately 93.0% reduced training and testing cost. It is observable from Table 3 that when eigenvectors with approximately 80%–90% cumulative variance are used, the sensitivity and specificity are decreased by 5.63% and 3.37% respectively compared to the proposed approach. This is due to presence of higher eigenvalued eigenvectors that increases false-positive and false-negative rate.

Data visualization is shown in Figure 5 for all datasets (for simplicity, few classes are considered for MNIST and Indian Pines). For MNIST dataset, classes '0', '3', '8' and '9' are considered to show the workability of the proposed method in presence of deceptive similar classes. For two dimensional visualization top two eigenvectors are utilized for following two scenarios: a) transformed feature space obtained by proposed method and b) transformed feature space obtained by considering the cumulative variance of approximately 80%–90%. From visualization, it is observable that the extracted features from the proposed approach efficiently represents the CoI with strong separation from other class samples. The sensitivity is inversely proportional to specificity and vice versa. Following this, a new performance measure parameter called stability factor (SF) is introduced to justify the robustness of the proposed method (shown in Eq. 18). The values of SF nearer to 1 ensures the stability of the propose model and validates the robustness of the model against false-positive and false-negative

Table 2. Parameter values and performance calculation with the proposed method.

S. no.	**Dataset**	**Class**	**Size of K**	γ	ν	**A**	**P**	**Sp**	**S**
1	Iris	Setosa	2	0.09	0.012	0.99	1.0	1.0	0.97
		Vergnica	2	1.2	0.015	0.99	1.0	1.0	0.99
		Versicolor	2	0.10	0.04	0.99	1.0	0.99	1.0
2	svmGuide1	0	2	0.005	0.043	0.95	0.96	0.98	0.98
		1	2	0.145	0.134	1.0	0.99	1.0	0.98
3	Credit Fraud	0	5	1.25	0.041	0.98	1.0	0.99	0.99
		1	2	1.30	0.081	0.99	1.0	0.99	1.0
4	Diabetes	0	3	6.163	0.05	0.99	1.0	1.0	0.99
		1	3	6.014	0.04	0.99	1.0	0.99	1.0
5	Glass	0	4	16.13	0.452	0.98	1.0	0.99	1.0
		1	3	12.34	0.045	0.93	0.96	0.98	0.99
		2	4	10.33	0.023	0.94	0.98	0.98	0.98
		3	4	14.31	0.432	0.97	0.96	0.98	0.98
		4	3	1.50	0.031	0.98	0.98	0.98	0.98
		5	4	1.55	0.241	0.98	0.99	0.98	0.98
6	Heart	0	4	10.33	0.153	0.98	0.97	0.95	1.0
		1	3	12.33	0.006	0.98	0.97	0.95	0.98
7	Wine	0	2	1.4	0.023	0.99	1.0	0.99	1.0
		1	3	1.2	0.015	0.99	1.0	0.98	1.0
		3	2	0.10	0.04	0.98	0.98	0.97	0.98
8	MNIST	0	42	0.08	0.001	0.98	1	0.99	1
		1	49	1.3	0.012	0.99	0.98	1.0	0.98
		2	43	0.048	0.142	0.98	0.98	0.99	0.98
		3	39	0.932	0.012	0.99	0.99	0.99	1.0
		4	41	1.9	0.015	0.99	0.99	0.98	0.99
		5	38	0.004	0.013	0.98	0.98	0.98	0.99
		6	39	0.045	0.131	0.98	1.0	0.98	0.99
		7	38	0.005	0.142	0.98	1.0	0.98	0.99
		8	37	0.082	0.001	0.99	0.99	0.98	0.99
		9	49	0.9	0.01	0.99	0.98	1.0	0.99
9	Indian Pine	Alfalfa	13	0.096	0.026	0.97	0.97	0.95	0.93
		Corn-notill	28	0.002	0.04	0.98	1.0	1.0	0.96
		Corn-mintill	29	0.006	0.032	0.93	1.0	1.0	0.86
		Corn	18	0.02	0.032	0.95	1.0	1.0	0.91
		Grass-pasture	29	0.02	0.313	0.94	1.0	1.0	0.87
		Grass-trees	16	0.01	0.04	0.98	1.0	1.0	0.96
		Grass-pasture-mowed	17	0.0006	0.004	0.96	0.93	0.99	0.95
		Hay-windrowed	19	6e-7	0.032	0.98	0.98	0.98	0.97
		Oats	21	2e-7	0.031	0.95	0.79	0.96	0.98
		Soybean-notill	29	0.09	0.035	0.96	1.0	0.96	0.99
		Soybean-mintill	44	0.08	0.037	0.93	1.0	1.0	0.91
		Soybean-clean	28	0.06	0.031	0.94	1.0	0.98	0.93
		Wheat	7	0.08	0.003	0.95	1.0	1.0	0.91
		Woods	12	0.089	0.014	0.89	0.84	0.88	0.98
		Buildings-Grass-Trees-Drives	24	0.008	0.032	0.95	1.0	0.98	0.99
		Stone-Steel-Towers	11	0.075	0.034	0.95	1.0	1.0	0.88

*A- Accuracy, P- Precision, Sp- Specificity, S- Sensitivity, γ and ν are hyperparameters of OCSVM.

Table 3. Performance comparison.

S. no.	Dataset name	Class	Original features				Transformed features with approximately 80%–90% variance				Transformed features with proposed method				Transformed features with approximately 50% variance			
			A	**P**	**Sp**	**S**	**A**	**P**	**Sp**	**S**	**A**	**P**	**Sp**	**S**	**A**	**P**	**Sp**	**S**
1	Iris	Setosa	0.99	1.0	1	0.96	0.99	1.0	1.0	0.96	0.99	1.0	1.0	0.97	0.99	0.98	0.98	0.95
		Vergnica	0.98	1.0	0.97	0.96	0.99	1.0	0.98	0.97	0.99	1.0	1.0	0.99	0.99	1.0	0.98	0.98
		Versicolor	0.98	0.98	0.97	0.96	0.98	0.98	0.98	0.96	0.99	1.0	0.99	1.0	0.99	0.98	0.9	0.98
2	svmGuide1	0	0.90	0.87	0.87	0.85	0.92	0.9	0.95	0.97	0.95	0.96	0.98	0.98	0.94	0.95	0.98	0.96
		1	0.97	0.99	0.99	0.97	0.98	0.98	1.0	0.99	1.00	0.99	1.0	0.98	0.98	1.0	0.98	0.98
3	Credit Fraud	0	0.98	0.97	0.99	0.96	0.98	0.98	0.98	1.0	0.98	1.0	0.99	0.99	0.96	0.95	0.91	0.95
		1	0.95	0.95	0.94	0.95	0.92	1.0	1.0	0.98	0.99	1.0	0.99	1.0	0.96	0.99	0.99	0.98
4	Diabetes	0	0.97	0.98	0.95	0.95	0.98	1.0	1.0	0.96	0.99	1.0	1.0	0.99	0.99	0.98	0.97	0.98
		1	0.97	1.0	1.0	0.96	0.97	1.0	1.0	0.96	0.99	1.0	0.99	1.0	0.98	0.95	0.98	0.97
5	Glass	0	0.93	0.88	0.95	0.94	0.92	0.85	0.95	0.93	0.98	1.0	0.99	1.0	0.97	0.95	0.98	0.97
		1	0.87	0.95	0.92	0.89	0.90	0.96	0.93	0.95	0.93	0.96	0.98	0.99	0.89	0.96	0.93	0.95
		2	0.91	0.98	0.94	0.91	0.91	0.97	0.94	0.89	0.94	0.98	0.98	0.98	0.91	0.98	0.94	0.95
		3	0.96	0.96	0.92	0.9	0.97	0.96	0.92	0.92	0.97	0.96	0.98	0.98	0.97	0.98	0.97	0.91
		4	0.97	0.95	0.9	0.93	0.96	0.95	0.9	0.9	0.98	0.98	0.98	0.98	0.97	0.95	0.9	0.96
		5	0.96	0.97	0.94	0.93	0.97	0.94	0.94	0.95	0.98	0.99	0.98	0.98	0.94	0.94	0.94	0.93
6	Heart	0	0.93	0.95	0.96	0.89	0.93	0.95	0.95	0.88	0.98	0.97	0.95	1.0	0.95	0.95	0.97	0.98
		1	0.96	0.93	0.94	0.91	0.95	0.93	0.91	0.92	0.98	0.97	0.95	0.98	0.97	0.93	0.91	0.95
7	Wine	0	0.99	1.0	0.99	0.97	0.99	1.0	0.99	0.97	0.99	1.0	0.99	1.0	0.99	1.0	0.99	0.98
		1	0.99	1.0	0.98	0.97	0.99	1.0	0.98	0.97	0.99	1.0	0.98	1.0	0.99	1.0	0.98	0.97
		2	0.98	0.98	0.97	0.96	0.99	0.98	0.98	0.97	0.98	0.98	0.97	0.98	0.95	0.96	0.95	0.98
8	MNIST	0	0.98	1.0	0.99	0.99	0.98	1.0	0.99	1.0	0.98	1.0	0.99	1.0	0.98	1.0	0.99	1.0
		1	0.98	0.95	0.98	0.93	0.94	0.93	0.95	0.91	0.99	0.98	1.0	0.98	0.98	0.94	0.99	0.98
		2	0.97	0.94	0.94	0.91	0.95	0.92	0.94	0.90	0.98	0.98	0.99	0.98	0.98	0.94	0.97	0.96
		3	0.98	0.95	0.89	0.96	0.97	0.96	0.87	0.91	0.99	0.99	0.99	1.0	0.94	0.94	0.89	0.93
		4	0.98	0.95	0.98	0.93	0.97	0.96	0.97	0.91	0.99	0.99	0.98	0.99	0.95	0.94	0.95	0.98
		5	0.95	0.93	0.98	0.95	0.96	0.93	0.98	0.95	0.98	0.98	0.98	0.99	0.98	0.93	0.99	0.95
		6	0.98	0.95	0.98	0.93	0.97	0.96	0.97	0.91	0.98	1.0	0.98	0.99	0.98	0.94	0.99	0.98
		7	0.98	0.95	0.98	0.93	0.97	0.93	0.94	0.91	0.92	1.0	0.98	0.99	0.93	0.94	0.99	0.88
		8	0.96	0.95	0.98	0.93	0.95	0.93	0.96	0.91	0.99	0.99	0.98	0.99	0.92	0.91	0.91	0.89
		9	0.98	0.95	0.98	0.93	0.97	0.96	0.97	0.91	0.99	0.98	1.0	0.99	0.94	0.98	0.94	0.95
9	Indian Pine	Alfalfa	0.85	0.97	0.97	0.75	0.86	0.98	0.98	0.8	0.98	0.98	0.99	0.97	0.9	1.0	0.99	0.88
		Corn-notill	0.96	0.96	0.97	0.9	0.97	0.97	0.95	0.93	0.98	1.0	1.0	0.98	0.98	0.97	0.97	0.96
		Corn-mintill	0.93	1.0	1.0	0.86	0.93	1.0	1.0	0.86	0.97	1.0	1.0	0.98	0.95	0.97	0.98	0.94
		Corn	0.93	1.0	1.0	0.90	0.93	1.0	1.0	0.92	0.97	1.0	0.98	0.94	0.95	0.98	0.98	0.9
		Grass-pasture	0.94	1.0	0.98	0.93	0.94	0.98	0.98	0.94	0.97	1.0	1.0	0.99	0.95	1.0	1.0	0.98
		Grass-trees	0.90	0.96	0.93	0.86	0.98	1.0	1.0	0.93	0.98	1.0	1.0	0.96	0.98	1.0	1.0	0.96
		Grass-pasture-mowed	0.9	0.84	0.8	0.93	0.91	0.87	0.81	0.97	0.98	0.98	0.99	0.97	0.93	0.86	0.85	0.93
		Hay-windrowed	0.9	0.88	0.89	0.93	0.94	0.92	0.90	0.98	0.98	0.98	0.98	0.97	0.93	0.89	0.88	0.95
		Oats	0.95	0.87	0.99	0.78	0.98	0.87	0.99	0.70	0.95	0.89	0.96	0.98	0.95	0.79	0.96	0.98
		Soybean-notill	0.96	1	0.96	0.98	0.97	1.0	0.97	0.98	0.96	1.0	0.98	0.99	0.96	1.0	0.96	0.99
		Soybean-mintill	0.93	0.92	0.91	0.89	0.94	1.0	1.0	0.9	0.95	1.0	1.0	0.94	0.93	1.0	1.0	0.92
		Soybean-clean	0.94	0.94	0.98	0.9	0.94	1.0	0.98	0.9	0.95	1.0	0.98	0.95	0.94	1.0	0.98	0.94
		Wheat	0.95	1.0	1.0	0.90	0.95	1.0	1.0	0.92	0.95	1.0	1.0	0.95	0.94	0.97	1.0	0.93
		Woods	0.89	0.84	0.88	0.93	0.9	0.84	0.90	0.93	0.94	0.92	0.94	0.98	0.92	0.85	0.88	0.98
		Buildings-Grass-Trees-Drives	0.88	0.83	0.87	0.91	0.89	0.84	0.89	0.98	0.95	1.0	0.98	0.99	0.90	0.88	0.89	0.99
		Stone-Steel-Towers	0.94	0.91	0.88	0.80	0.93	1.0	0.98	0.8	0.95	1.0	1.0	0.92	0.95	1.0	1.0	0.86
Average Performance			0.95	0.96	0.96	0.92	0.93	0.96	0.96	0.93	0.97	0.99	0.99	0.98	0.95	0.96	0.96	0.95

*A- Accuracy, P- Precision, Sp- Specificity, S- Sensitivity

predictions. Figure 6 shows that the proposed approach is more stable compared to reduced feature space with approximately 80%–90% cumulative variance.

$$SF = \frac{Specificity_{T_c}}{Sensitivity_{T_c}} \tag{18}$$

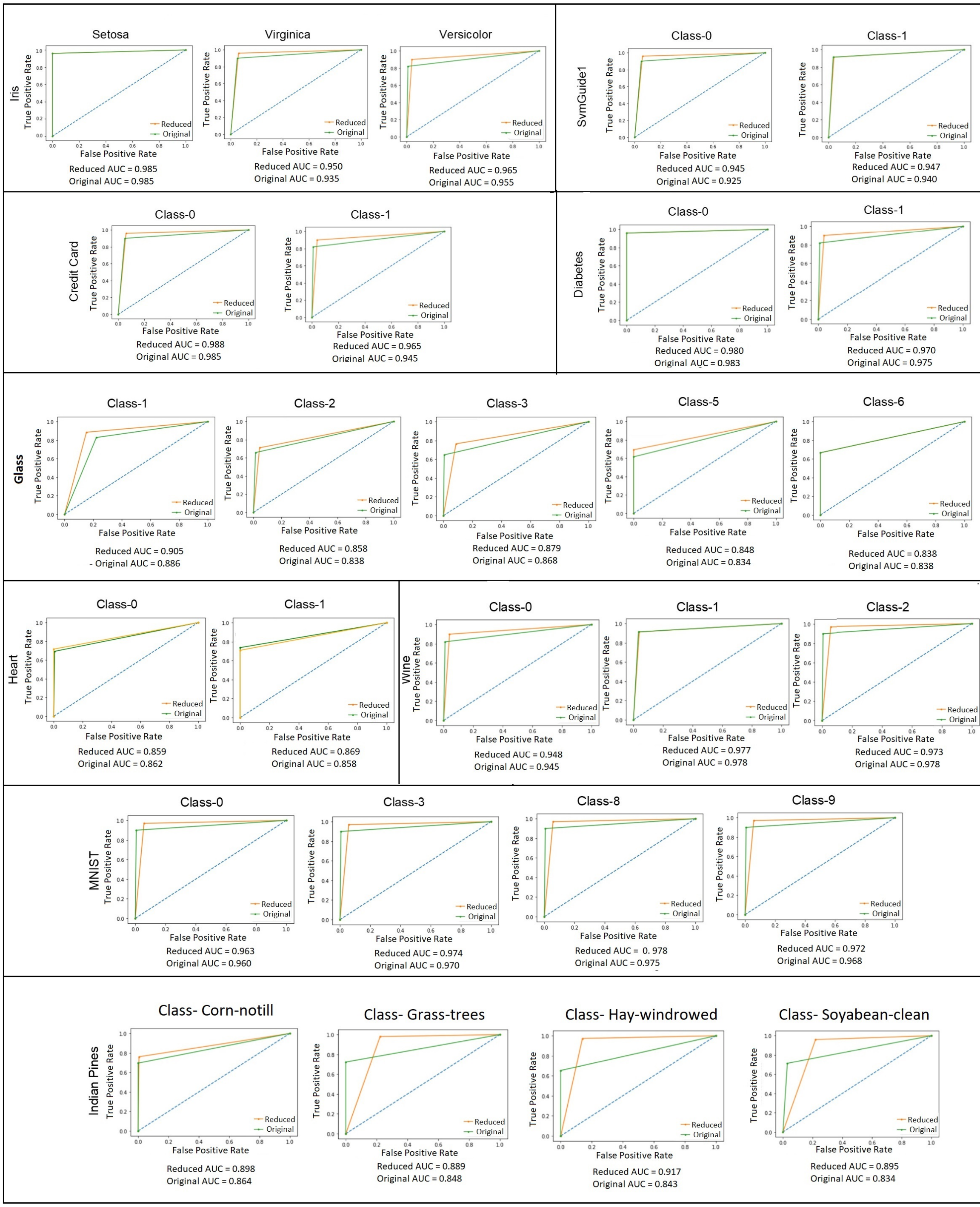


Figure 4. AUROC curve for target classes of all datasets. (shows the comparative performance of proposed method over original features).

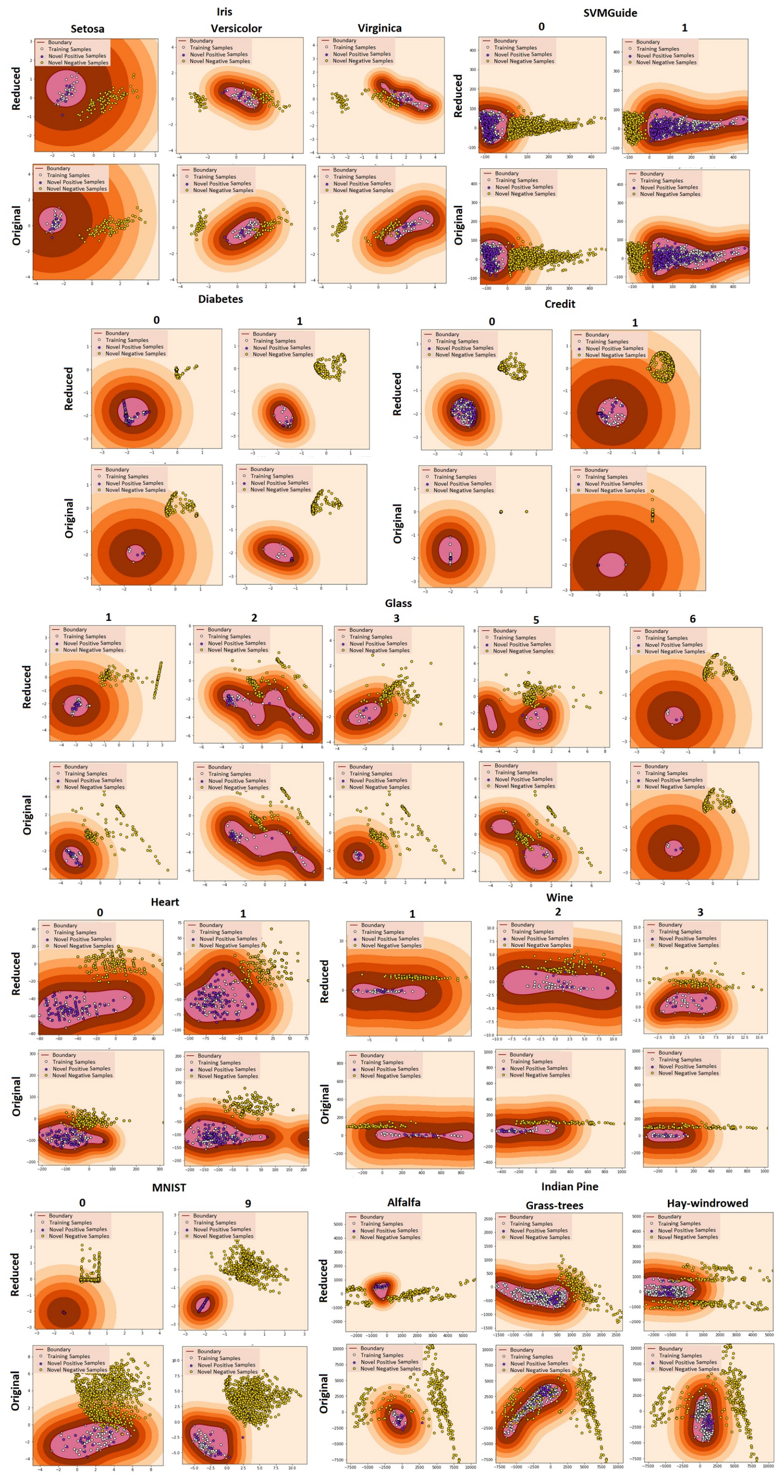
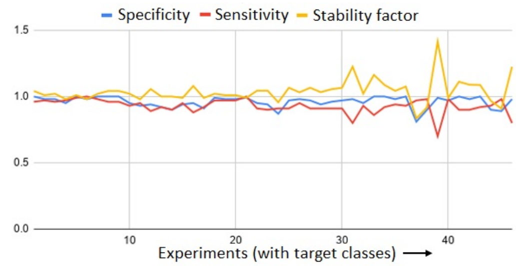
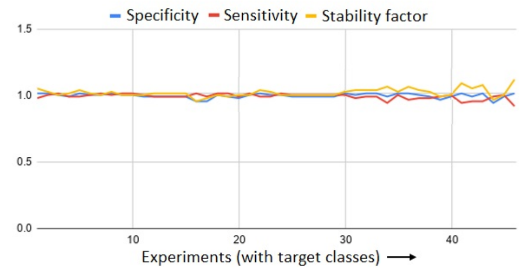


Figure 5. Two dimensional visualization of all datasets with PCs obtained with original features and proposed method. (shows strong separation of target class).



(a) Performance of features with approximately 80%–90% variance.



(b) Performance of features with proposed approach.

Figure 6. Specificity, sensitivity and stability factor. (shows the robustness of the proposed model).

5. Conclusion

The present research proposes a novel approach of target class guided feature transformation to ensure the strong separation of class-of-interest samples from other classes. The proposed algorithm is named as learning target class eigen subspace (LTS-ES). In this context; initially, the eigen decomposition is performed on the target class samples, and later, the higher and lower eigenvalued eigenvectors of variance-covariance matrix are rejected based on the cumulative captured variance. A novel way of rejecting the least important and misleading eigenvectors is proposed after extensive experiments because the higher eigenvalued eigenvectors carry high variance that may cause overfitting, whereas lower eigenvalued eigenvectors do not contribute significant information of the target class. The proposed approach of selection of promising eigenvector is found suitable for all the datasets. The selected eigenvectors are treated as knowledge grid to transform the training and testing samples. The experiments have been performed in presence of: original features, transformed features with eigenvectors containing approximately 80%–90% cumulative variance, transformed features with proposed approach and transformed features with eigenvectors containing approximately 50% cumulative variance. Further, through two dimensional visualization the effectiveness of the proposed model is shown compared to the transformed features obtained by higher eigenvalued eigenvectors with approximately 80%–90% cumulative variance, where the proposed model efficiently projects the other class samples away from the target class. It is also evident from experiments that the average number of features obtained by the proposed model is approximately 7% of the original feature space, and these transformed features outperformed over original features and transformed features obtained using higher eigenvalued knowledge grid. Compared to other feature selection scenarios in this research, the proposed method achieves increase in average accuracy by 1.87%–4.27%, precision by 2.90%–3.09%, specificity by 2.62%–2.82% and sensitivity by 2.88%–6.76% with approximately 93.0% reduced training and testing cost. The present research work can be further extended for distributed environment, where the temporal arrival of samples or batches of samples with massive features leads computational overhead.

References

[1] Minter T. Single-class classification. In: LARS Symposia; 1975. p. 54.

[2] Alam S, Sonbhadra SK, Agarwal S, Nagabhushan P. One-class support vector classifiers: A survey. Knowledge-Based Systems. 2020:105754.

[3] Chalapathy R, Chawla S. Deep learning for anomaly detection: A survey. arXiv preprint arXiv:190103407. 2019.

[4] Pimentel MA, Clifton DA, Clifton L, Tarassenko L. A review of novelty detection. Signal Processing. 2014;99:215-49.

[5] Bellman R. Dynamic programming. Princeton University Press, New Jersey. 1957;8.

[6] Alam S, Sonbhadra SK, Agarwal S, Nagabhushan P, Tanveer M. Sample reduction using farthest boundary point estimation (FBPE) for support vector data description (SVDD). Pattern Recognition Letters. 2020;131:268-76.

[7] Johnstone IM, Titterington DM. Statistical challenges of high-dimensional data. The Royal Society Publishing; 2009.

[8] Sonbhadra SK, Agarwal S, Nagabhushan P. Learning Target Class Feature Subspace (LTC-FS) Using Eigenspace Analysis and N-ary Search-Based Autonomous Hyperparameter Tuning for OCSVM. International Journal of Pattern Recognition and Artificial Intelligence. 2021:2151015.

[9] Houari R, Bounceur A, Kechadi MT, Tari AK, Euler R. Dimensionality reduction in data mining: A Copula approach. Expert Systems with Applications. 2016;64:247-60.

[10] Pal M, Foody GM. Feature selection for classification of hyperspectral data by SVM. IEEE Transactions on Geoscience and Remote Sensing. 2010;48(5):2297-307.

[11] Duch W. Filter methods. In: Feature Extraction. Springer; 2006. p. 89-117.

[12] Kohavi R, John GH. Wrappers for feature subset selection. Artificial intelligence. 1997;97 (1-2):273-324.

[13] Van Der Maaten L, Postma E, Van den Herik J. Dimensionality reduction: a comparative. J Mach Learn Res. 2009;10 (66-71):13.

[14] Huang X, Wu L, Ye Y. A Review on Dimensionality Reduction Techniques. International Journal of Pattern Recognition and Artificial Intelligence. 2019;33 (10):1950017.

[15] Gauch Jr HG. Noise reduction by eigenvector ordinations. Ecology. 1982;63 (6):1643-9.

[16] Josse J, Husson F. Selecting the number of components in principal component analysis using cross-validation approximations. Computational Statistics & Data Analysis. 2012;56 (6):1869-79.

[17] Kim PJ, Chang HJ, Choi JY. Fast incremental learning for one-class support vector classifier using sample margin information. In: 2008 19th International Conference on Pattern Recognition. IEEE; 2008. p. 1-4.

[18] Chandrashekar G, Sahin F. A survey on feature selection methods. Computers & Electrical Engineering. 2014;40 (1):16-28.

[19] Björklund M. Be careful with your principal components. Evolution. 2019;73 (10):2151-8.

[20] Nguyen LH, Holmes S. Ten quick tips for effective dimensionality reduction. PLoS computational biology. 2019;15 (6).

[21] Ferré L. Selection of components in principal component analysis: a comparison of methods. Computational Statistics & Data Analysis. 1995;19 (6):669-82.

[22] Choi Y, Taylor J, Tibshirani R. Selecting the number of principal components: Estimation of the true rank of a noisy matrix. The Annals of Statistics. 2017;45 (6):2590-617.

[23] Peres-Neto PR, Jackson DA, Somers KM. How many principal components? Stopping rules for determining the number of non-trivial axes revisited. Computational Statistics & Data Analysis. 2005;49 (4):974-97.

[24] Tax DM, Müller KR. Feature extraction for one-class classification. In: Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003. Springer; 2003. p. 342-9.

[25] Heskes T. Bias/variance decompositions for likelihood-based estimators. Neural Computation. 1998;10 (6):1425-33.

[26] Cho SB. Recognition of unconstrained handwritten numerals by doubly self-organizing neural network. In: Proceedings of 13th International Conference on Pattern Recognition. vol. 4. IEEE; 1996. p. 426-30.

[27] Lian H. On feature selection with principal component analysis for one-class SVM. Pattern Recognition Letters. 2012;33 (9):1027-31.

[28] Feng K, Jiang Z, He W, Ma B. A recognition and novelty detection approach based on Curvelet transform, nonlinear PCA and SVM with application to indicator diagram diagnosis. Expert Systems with Applications. 2011;38 (10):12721-9.

[29] Jeong YS, Kang IH, Jeong MK, Kong D. A new feature selection method for one-class classification problems. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews). 2012;42 (6):1500-9.

[30] Nagabhushan P, Meenakshi H. Target class supervised feature subsetting. International Journal of Computer Applications. 2014;91 (12):11-23.

[31] Sonbhadra SK, Agarwal S, Nagabhushan P. Early-stage COVID-19 diagnosis in presence of limited posteroanterior chest X-ray images via novel Pinball-OCSVM. arXiv preprint arXiv:201008115. 2020.

[32] Tax DM, Duin RP. Support vector domain description. Pattern recognition letters. 1999;20 (11-13):1191-9.

[33] Schölkopf B, Platt JC, Shawe-Taylor J, Smola AJ, Williamson RC. Estimating the support of a high-dimensional distribution. Neural computation. 2001;13 (7):1443-71.

[34] Sonbhadra SK, Agarwal S, Nagabhushan P. Target specific mining of COVID-19 scholarly articles using one-class approach. Chaos, Solitons & Fractals. 2020;140:110155.

[35] Erfani SM, Rajasegarar S, Karunasekera S, Leckie C. High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. Pattern Recognition. 2016;58:121-34.