Computing and Artificial Intelligence 2026, 4(2), 4251.
doi: https://doi.org/10.59400/cai4251

Article

A generalized instance weighting principal component analysis framework

Ernest Domanaanmwi Ganaa

Department of Information and Communication Technology, Faculty of Applied Science and Technology, Dr. Hilla Limann Technical University, Wa P.O. Box 553, Ghana; eganaa@dhltu.edu.gh

CITATION

Ganaa ED. A generalized instance weighting principal component analysis framework. Computing and Artificial Intelligence. 2026; 4(2): 4251. https://doi.org/10.59400/cai4251

ARTICLE INFO

Received: 10 February 2026
Revised: 5 May 2026
Accepted: 8 May 2026
Available online: 26 May 2026

Abstract: As a classical algorithm, principal component analysis (PCA) considers all data points as equally important in pursuing projections, although some data points may be more useful than others. This consequently makes traditional PCA sensitive to noisy or corrupt data. Another challenge faced by PCA is the problem of dealing with missing observations. These two challenges may significantly affect its performance negatively. In this study, we present a generalized instance weighting PCA (IWPCA) algorithm that innovatively addresses the problems of the traditional PCA outlined above. Instead of seeking orthogonal projections directly in PCA, data is embedded in the projection space to remove a random scaling factor in the projection space and therefore, solves the issue of the missing observations challenge. To achieve better projections, a sample instance is further introduced into the model. Two approaches are then used to weigh instances from geometric and data learning views. Thus, minimizing the impact of outliers and noisy instances in the proposed algorithm. This makes the proposed model a special form of the classical PCA where instances are weighed in pursuing projections, thus improving performance by enhancing projections. Extensive experimental evaluations indicate the proposed method has superior performance over all the comparative methods.

Keywords: principal component analysis (PCA); instance weighting; dimensionality reduction

1. Introduction

In machine learning, dimensionality reduction inevitably plays a very crucial role in classification and clustering tasks since these are done more efficiently and effectively in compact subspaces than in the original high-dimensional spaces. Dimensionality reduction [1–4] is the conversion of high-dimensional data into a low dimensionality with minimal information lost. Several dimension reduction techniques have been advanced by researchers over the years. Principal component analysis (PCA) [3] is one such dimension reduction algorithm. Locality sensitive discriminant analysis (LSDA) [5, 6] discovers a projection that takes full advantage of the margin between instances from dissimilar categories at each local area. Considering that the linear relationship between samples will likely change in a low-dimensional space, but linear embedding methods try to preserve it, and this may negatively affect recognition accuracy, a regularized discriminant method [7] was presented to address this issue. Locality preserving projections (LPP) [8, 9] also discover a good linear embedding that preserves intrinsic structure information. Wang et al. [10] proposed a graph embedding algorithm that concurrently learns the anchors, the membership, projection

matrices, and global structures with the incorporation of PCA. Additionally, Zhang et al. [11] proposed generalized non-convex regularizers that accommodate several surrogate functions that approximate the tensor rank function and l_0-norm.

Research in dimensionality reduction is still vigorously explored by researchers in a bid to improve the performance of existing methods or present novel ones. Among these dimensionality reduction techniques, PCA seems to be the most widely applied technique. The loss function of PCA is presented below:

$$\arg\min_{w^T w=1} \sum\nolimits_{i=1}^{n} (x_i - ww^T x_i)^2 = \left\|X - ww^T X\right\|_F^2 = \arg\ \max \left\|X^T w\right\|_2^2 \quad (1)$$

$\{w_j\}_{j=1}^{d}$ is a subclass of orthogonal projection vectors in $\Re^m$ and the set of data points $\{x_i\}_{i=1}^{n}$ is zero-mean m-dimensional data points. The above function minimizes the least squares reconstruction error.

The traditional PCA certainly performs well given clean data. However, because of its sensitivity to missing and outlying observations [12,13], the presence of these in datasets may negatively affect its performance. Even more concerning is the fact that these corrupt data points are becoming very prevalent in datasets which could be due to a couple of reasons such as sensor failures, illuminations, bad poses, occlusions in the foreground amongst others. Due to these two challenges and others, different variations of PCA have been presented to address such challenges.

Some of these variants of PCA include Robust PCA (ROBPCA) [14] that minimizes the effect of noisy samples by integrating robust covariance estimation and projection pursuit. Fast Iterative Robust (FIR) PCA [15] efficiently estimates the inliers' center location and covariance. Nie et al. [16] put forward a variant of PCA that automatically removes the mean in datasets. Jana et al. [17] in seeking to reduce the impact of noise on algorithms replaced the l_p norm with the l_0 norm. A similar algorithm [18] improves how the dense basis of PCA is interpreted by discovering a sparse basis. Jenatton et al. [19] also proposed another adaptation of sparse PCA called structured sparse PCA (SSPCA). Jiang et al. [20] presented a version of PCA called gLPCA that discovers a low-dimensional structure of the data by integrating graph structures. In the study by Kumar et al. [21], two PCA techniques that are based on the idea of cancellable biometrics that seek to prevent biometrics from being compromised by intruders were proposed.

Datasets are incomplete and not perfect since they are made up of missing observations and noise. As a result, algorithms must find effective ways of handling such data. However, most existing robust methods engage in suitable feature selection in pursuing projections, but we think that the most effective way to deal with the negative impact of noise is to rather engage in suitable sample selection. It is therefore necessary to identify the instances that are the most relevant in pursuing optimal projections. In this study we therefore present a technique that learns the underlying low-dimensional structure that compactly characterizes the dataset amid missing observations and noise. Thus, our algorithm will significantly reduce the negative impact of noise and missing observations on computations.

Based on the ongoing discussions, we present a new variant of PCA called a

generalized Instance Weighting PCA framework (IWPCA) for image recognition to solve the missing observation and outlier challenges of the traditional PCA from a different perspective. By expanding the traditional PCA strategy of minimizing least squares reconstruction error in Equation (1) above, we incorporate data in pursuing orthogonal projections as the first idea of this paper. As demonstrated in **Figure 1**, **Figure 1a** is an illustration of a two-dimensional unit projection vector in PCA. More reasonable projection vectors should fit into the data itself. Therefore, our ideal projection vector can be obtained with data being embedded in the projection space, which can be observed in **Figure 1b**. Through observation, the difference between **Figure 1b** and **Figure 1a** is that **Figure 1b**'s constraint integrates the projection vector and data, whereas **Figure 1a** does not. This integration of the projection and data in the constraint embeds the data in the projection space in our model. This addresses the missing observation challenge of an arbitrary scaling factor in the projection space since only observed data would be embedded, leaving out the missing ones.

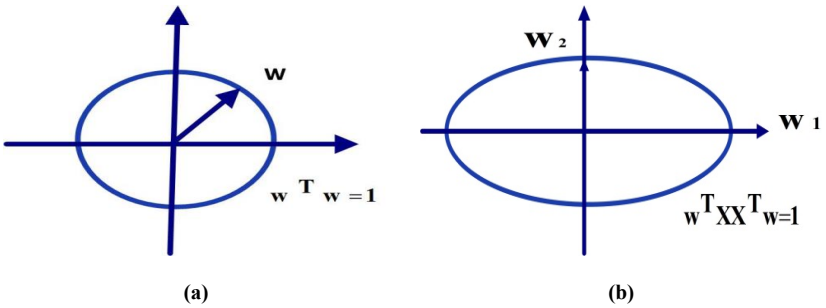


Figure 1. Illustration of **(a)** a unit projection vector in PCA, **(b)** a data-embedded projection vector in our algorithm.

The above discussion inspires us to examine the relevance of each data point using two approaches in this paper. Thus, our model treats data points differently since some data samples may be more useful or representative than others, thereby enhancing the model's discriminative abilities. In our first strategy, we consider the overall distance of each data point from the remaining instances in the training set. The bigger the overall distance of an instance from the other training examples, the more likely it is an outlier and hence likely to be less important.

Secondly, we use gradient measurement via data learning. The data importance is learned directly through gradient descent. We iteratively refine the weights of the training set through gradient descent. Through this iterative refining process, we can further obtain optimal projections.

By evaluating the importance of each instance, the proposed framework learns the intrinsic data structure of the entire dataset. Generally, our algorithm learns enhanced projections by distinguishing between noise and good data points through these two strategies to improve its performance. A summary of the contributions of this study includes: We present a new framework for dealing with both missing observations and outlier detection challenges in traditional PCA from a new perspective by considering data distribution and instance weighting in one model. We further propose two different strategies for evaluating instance importance: distance and gradient measurements, details of which are presented in sub-sections of Section 3. Experimental evaluations

indicate our approach is effective.

The rest of the study is structured into 5 sections. Sections 2 and 3 take care of related works and formulation of the proposed algorithm, respectively. Results are presented and discussed in Section 4 while Section 5 deals with conclusion and future works.

2. Related works

Principal component analysis and its variations are reviewed in this section.

2.1. PCA and its variants

Since we have lost control over the data acquisition process coupled with the sensitive nature of traditional PCA to outliers or noisy data and missing observations, several variations of PCA have been developed. Despite these several adaptations to address the above challenges in the traditional PCA, these challenges persist.

One such variant of PCA is a generative robust PCA that deals with data noise as a mixture of Gaussians using the Bayesian framework [22]. Tabaghi et al. [23] in seeking to find a compact subspace that best represents a set of manifold data points with the minimum average cost of projecting such data points onto the compact subspace, proposed another variant of PCA. Also, pivot-aware PCA (PAPCA) [24] collaboratively reduces the effects of noisy samples and Block-PCA [25] improves its robustness by exploiting both latent and spatial structural information. Another adaptation of PCA which was proposed to fix the outlier shortfall of the traditional PCA is ROBPCA [26]. This method combines projection pursuit and robust covariance approximation. Xu et al. [27] also proposed outlier PCA: the high-dimensional case which is robust to contaminated data. Generalized spherical principal component analysis (GSPCA) [28] is a robust version of PCA grounded on the generalized spatial sign covariance matrix. Max-min robust PCA via binary weights [29] innovatively combines reconstruction error and projection variance to accurately learn the projection matrix.

Similarly, Zhan and Vaswani [30] proposed robust PCA with partial subspace knowledge; Zhou and Feng [31] also proposed outlier-robust tensor PCA (OR-TPCA); in the study by Higuchi and Eguchi [32], a robust PCA with adaptive selection for tuning parameters was presented and Candes et al. [33] also proposed robust PCA. Other similar versions include the literature [34] which is a robust PCA for skewed data and its outlier map; a mixed-noise removal PCA method for hyperspectral images was proposed [35].

Besides the robust variants of PCA, sparse PCA is another adaptation. Such variants include SDPCA [36] which is a discriminative PCA version that considers that there exist differences between data points. Shen and Huang [37] also proposed sparse PCA through regularized SVD (sPCA-rSVD). Similarly, Chen and Sun [38] proposed l21 norm-based sparse PCA with the trace norm regularized term to learn both the optimal projection matrix and optimal mean simultaneously. Another sparse PCA that provides certifiable optimality at the scale of choosing $k = 5$ covariates from $p = 300$ variables has been proposed [39]. GeoSPCA [40] while satisfying the orthogonality constraints builds all the principal components at once. By incorporating the l21 norm

minimization into an objective function, this makes the regression coefficients sparse and robust to noisy instances. Like GeoSPCA [40], JSPCA [41] selects useful features to enhance its robustness to outliers. Also, weighted SPCA [42] assigns weights to the elements of a residual matrix. Del Pia [43] solves the computational complexity problem of sparse PCA using the rank of the covariance matrix.

Furthermore, there are also deflated versions of PCA where the covariance matrix is shrunk per iteration [44–48]. Others include SPCA-SP [49] which maintains a balance amongst sparsity, orthogonality, explained variance, and computational cost. Shi et al. [50] presented a kernel PCA that sequentially computes each row of the kernel matrix through iteration. PCA has undergone a dramatic revolution in the past few years; this has produced several of its variants, some of which are outlined above.

Other algorithms include DCNN [51] which uses an improved Chimp Optimization Algorithm to find an optimal DCNN architecture by making changes based on the baseline ChOA. Kosarirad et al. [52] proposed an algorithm that uses the Grasshopper Optimization Algorithm to choose optimal features in big data.

Despite the successes of existing algorithms, the proliferation of datasets with outliers and missing values due to the unguided nature of acquiring these datasets makes it important to continuously present robust algorithms for their analysis. Additionally, most previous methods fail to consider the individual importance of data points and missing values in pursuing projections.

2.2. Concept of missing observations and outliers

Since the data collection process is usually uncontrolled, certain values are normally absent in datasets due to errors in the collection process. Another phenomenon is the presence of noise or outliers in datasets. Outliers or noise are extreme values that significantly exhibit different characteristics from the rest of the data points [53]. In ML, missing data and outliers are two common challenges that can significantly impact the performance of models negatively [54].

It is therefore very crucial to handle these two phenomena (missing data and outliers) in ML to improve performance. In the case of missing values, these could be handled by dropping entire rows with missing values, imputations by replacing missing values with statistical measures or predicting such missing values. Additionally, algorithms such as the proposed method could be developed to handle missing values.

The case of outlying data points has led to the proposition of several robust algorithms to improve performance in the presence of outliers. The impact of missing values and outliers on ML models includes reduced accuracy, increased variance, bias and poor generalization.

3. The generalized instance weighting PCA (IWPCA) framework

We present details of our algorithm in the subsequent sub-sections.

The proposed IWPCA method

The proposed framework (IWPCA) aims to solve the missing observations and outlier challenges of the traditional PCA by embedding data in the projection space and

considering instance weighting, respectively in one model, given that data is not perfect. Thus, by incorporating these two ideas into the proposed framework, we simultaneously solve the above two challenges faced by the traditional PCA.

The traditional PCA seeks orthogonal projections directly to minimize least squares reconstruction error. However, since PCA is a data-dependent projection scheme, it is sensitive to missing observations of data. In our proposed method, data is embedded to remove a random scaling factor in the projection and therefore solves the issue of missing observations. We incorporate each instance to scale its relevance in Equation (1) and get the following Equation (2):

$$\arg\min_{p^T p=1} \left\| X - Xpp^T \right\|_F^2 \tag{2}$$

$p = X^T w$ is a generalized orthogonal vector that considers its orthogonality in data distribution. This means our constraint that considers data distribution by embedding the data in the projection space can be re-written as $w^T X X^T w = 1$. By incorporating the data into this constraint, we force reasonable or most representative data to lie in the projection space. High-dimensional data lie in a low-dimensional space; by constraining the most representative data in the projection space, unreasonable or missing data will not be part of computations. This eliminates the issue of the proposed algorithm assigning random values or factors to missing values through imputations by replacing missing values with statistical measures or predictions. Such imputed values may have a negative impact on its performance. The above constraint is enforced in Equation (5). Thus, the issue of missing observations in the traditional PCA is solved by the above solution of considering orthogonal projections in the data distribution.

Practically, data is not perfect and contains outliers or noisy data points. These outliers and noisy data points can degrade the overall performance of applications. In our proposed method, we further weigh the importance of each data point by extending Equation (2) as in Equation (3).

$$\begin{aligned} &\min_{\widehat{p}} \left\| X - X\widehat{p}\widehat{p}^T \right\| \frac{2}{F} \\ &\text{s.t. } \widehat{p}^T \widehat{p} = 1 \end{aligned} \tag{3}$$

where $\widehat{p} = Dp$ is a generalized orthogonal vector and D = $\begin{bmatrix} d_1 & & \\ & \ddots & \\ & & d_n \end{bmatrix}$ is a diagonal matrix that assesses the significance of each data point in dataset X. We thus obtain the importance of each instance as follows:

$$\begin{bmatrix} d_1 & & \\ & \ddots & \\ & & d_n \end{bmatrix} \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} = \begin{bmatrix} d_1 x_1^T \\ \vdots \\ d_n x_n^T \end{bmatrix}$$

From the above, $\begin{bmatrix} d_1 x_1^T \\ \vdots \\ d_n x_n^T \end{bmatrix}$ becomes sample clean data without noise due to the imposition of the scaling factor d_i on each instance x_i which then enhances the sample

features w.

Equation (3) can also now be rewritten as below:

$$\begin{aligned} &\min_{D,p} \left\| X - XDpp^T D \right\|_F^2 \\ &p^T D^2 p = 1 \end{aligned} \tag{4}$$

Substituting $p = X^T w$ into Equation (4), we can convert the formula into:

$$\begin{aligned} &\arg\max_{D,w} w^T XDX^T XDX^T w \\ &\text{s.t. } w^T XDDX^T w = 1 \end{aligned} \tag{5}$$

Since $\widehat{p} = DX^T w$, we can see that the imposition of D, which is a penalty factor on the sample space will successively induce the sample feature space w. By introducing the Lagrange multiplier (λ) and taking partial derivatives of Equation (5) with respect to w to enforce the above constraint, we obtain Equation (6) as follows:

$$XDX^T XDX^T w = \lambda XDDX^T w \tag{6}$$

Equation (6) is a standard eigenvalue problem.

It can also be observed from the constraint that $w^T XDDX^T w = \sum_i^n (w^T x_i d_i)^2 = 1$.

With the assumption that datasets are incomplete and not perfect, d_i is a weight factor imposed on each instance to obtain refined instances without noise which then induces the sample features to improve the general performance of the proposed framework. Through the imposition of d_i on the sample space, the model can learn the intrinsic structures of datasets to control the effect of noisy instances to enhance the projections. Therefore, by considering data importance in our proposed model, we achieve a different model from traditional PCA. Thus, the proposed model is viewed as a special form of the traditional PCA with data distribution and instance weighting considered.

Obtaining matrix D (diagonal matrix)

We use two approaches to obtain the diagonal matrix D that estimates the significance of each data point in projections pursuit. These approaches are distance and gradient metrics. Details of these metrics are discussed in the next subsections.

(1) Distance measurement

From a geometrical view, the importance of a data point can be measured by its total distance, which is defined as the distance from an instance to the rest of the instances. The total distance of any data point from the rest of the training examples is a good way to measure its significance. **Figure 2** below illustrates this idea. As shown in **Figure 2**, we can detect that the total distance of data point x_i is larger than the total distance of each data point in the cluster. Thus, x_i is more likely to be a noisy data point and its importance will be scaled down to minimize its impact on the model. The algorithm for this distance measurement is shown in **Algorithm 1** below.

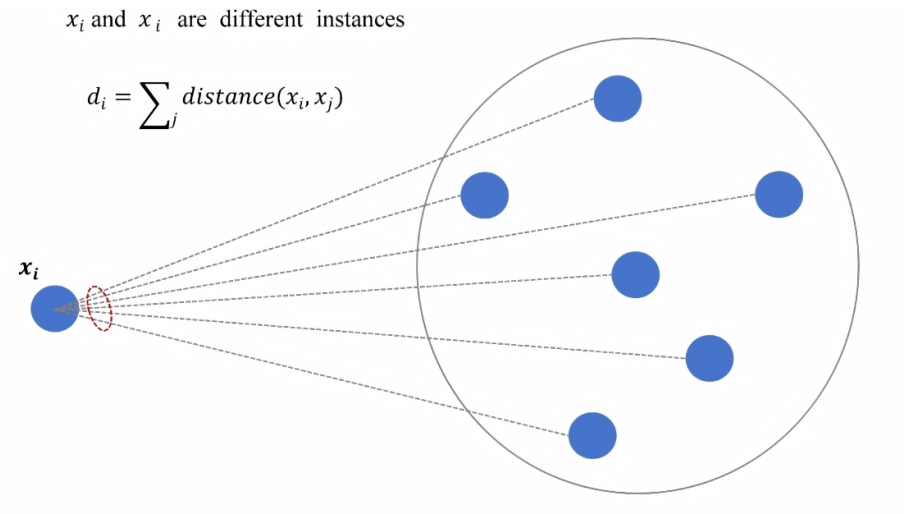


Figure 2. Demonstration of distance measurement of an instance.

Algorithm 1 Distance measurement in pursuing projection in our proposed model

1. Input: Training set X
2. Build importance evaluation diagonal matrix D
3. Obtain w based on Equation (6)
4. Output: The projection vector w

(2) Gradient measurement

Finally, in this measurement, the matrix D is learned from data directly via gradient descent. Considering $w^T X D X^T = \sum_i \mathrm{d}_i w^T x_i x_i^T$, we rewrite Equation (6) in its Lagrange form:

$$\sum_i \mathrm{d}_i w^T x_i x_i^T \sum_j \mathrm{d}_j x_j x_j^T w + \lambda \left(1 - \sum_i \mathrm{d}_i^2 w^T x_i x_i^T w\right) \tag{7}$$

After taking partial derivatives of Equation (7) and forcing it to be zero, we now have

$$\mathrm{d}_i w^T x_i x_i^T x_i x_i^T w + \sum_{j \neq i} \mathrm{d}_j w^T x_i x_i^T x_j x_j^T w - \lambda \mathrm{d}_i w^T x_i x_i^T w = 0. \tag{8}$$

The solution to Equation (8) is as follows:

$$\mathrm{d}_i = \frac{\sum_{j \neq i} \mathrm{d}_j w^T x_i x_i^T x_j x_j^T w}{w^T x_i (\lambda I - x_i^T x_i) x_i^T w}. \tag{9}$$

And finally, we use the gradient descent method to update D:

$$D \leftarrow D - \gamma \nabla D, \tag{10}$$

where γ is the learning rate of the model. A diagram demonstrating this update process of D using gradient descent is demonstrated in **Figure 3**. These updates refine D until the model converges. The learning rate (γ) controls the step size. The algorithm for the gradient measurement is as shown in **Algorithm 2** below.

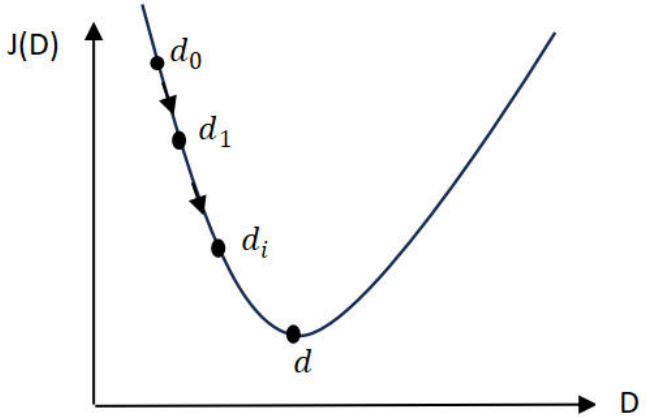


Figure 3. Demonstration of update of D using gradient descent.

Algorithm 2 Gradient measurement in pursuing projection in our proposed model

Input: Dataset X
Output: w
Parameter: Υ
Initialize:
-initialize diagonal matrix D as an identity matrix
While not converged do
1. Obtain w based on Equation (6)
2. For each data point x_i
3. Compute d_i based on Equation (9)
4. End for
5. Update Diagonal D
6. Compute loss from Equation (5)

4. Experimental evaluation

To validate the efficacy of our two algorithms: distance (IWPCA-1) and gradient (IWPCA-2) measurements, they are compared with eight (8) dimensionality reduction methods. These eight comparative methods include: PCA [3], LSDA [5], RCDA [7], LPP [9], FAGPP [10], RPCA-OM [16], gLPCA [20] and SDPCA [36]. Amongst these algorithms, PCA [3], LSDA [5] and LPP [9] are benchmark methods whilst the remaining five comparative algorithms are state-of-the-art ones.

We use six standard datasets: MNIST, USPS, ORL, FERET, UMIST and COIL-20 to conduct our experiments. Among these datasets, the first two are digit datasets, the third to fifth are face datasets and the last is an object dataset. These datasets were selected based on their different properties and features.

The MNIST (https://www.kaggle.com/datasets/hojjatk/mnist-dataset) dataset is a well-known dataset that has 60,000 training samples and 10,000 test sample images of handwritten digits from 0–9, with a resolution of 28 × 28 pixels with 256 levels of gray. The USPS (https://www.kaggle.com/datasets/bistaumanga/usps-dataset) handwritten dataset contains 7,291 training examples and 2,007 test examples of handwritten digits from 0–9. The ORL (https://paperswithcode.com/dataset/orl) face dataset has 40 subjects, each with 10 faces at different poses, making a total of 400 images of size 112 × 92. These images were taken at different times, lighting and facial expressions.

The faces are in an upright position in frontal view with a slight left-right rotation. The FERET (https://en.wikipedia.org/wiki/FERET_database) dataset contains 14,126 facial images comprising 1,199 individuals along with 365 duplicate sets of images that were taken on a different day. The UMIST (https://face.kairos.com/blog/60-facial-recognition-databases) dataset consists of 575 face images with a size of 112 × 92. The number of face images per person varies from 19 to 48 under different poses. This dataset can be obtained at https://doi.org/10.6084/m9.figshare.33130856.

The COIL-20 (https://www.bibsonomy.org/bibtex/2e21afb22e024792723fc3b9f659c522e/jabreftest) object dataset contains 1,440 observations of 20 objects with 72 poses each. The objects were placed on a motorized turntable against a black background and the turntable was rotated 360° to vary the object pose with respect to a fixed camera. Images of the 20 objects were taken at pose intervals of 5°.

4.1. Parameter settings

We run 21 experiments on each of the four datasets using each method and the average classification accuracies and standard deviations for all algorithms are computed. These average accuracies, standard deviations and best dimensions for the various algorithms are then recorded. All parameters of the comparative techniques are set according to their respective literature. For IWPCA-2, which represents the gradient descent method, the learning rate γ was selected from the range $0 < \gamma < 1$. Thus, $\gamma = \{0.1,\ 0.2,\ 0.3,\ 0.4,\ 0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9\}$. This is manually tuned for the gradient descent algorithm (IWPCA-2). The K-nearest neighbor () classifier was used for classification and therefore, parameter K was set to 5 for all methods.

4.2. Discussion of results and analysis

The results attained by all the methods on all four datasets are discussed below.

4.2.1. Experimental results discussion

The average classification accuracies with standard deviations and corresponding best dimensions of the different algorithms on all the datasets are presented in **Table 1** below.

Table 1. Average classification accuracies (%), standard deviations and optimal dimensions of all algorithms on all four datasets.

Algorithm	MNIST	USPS	ORL	COIL-20	Feret	UMIST
IWPCA-1	94.85 ± 0.02 (32)	**97.93 ± 0.40 (33)**	95.28 ± 0.13 (39)	98.34 ± 0.47 (17)	**69.20 ± 0.78 (203)**	98.86 ± 0.33 (30)
IWPCA-2	**96.00 ± 0.16 (30)**	97.42 ± 0.53 (39)	**96.42 ± 0.50 (24)**	**98.84 ± 0.05 (15)**	68.82 ± 0.85 (265)	**99.19 ± 0.32 (42)**
gLPCA	89.25 ± 0.39 (69)	95.31 ± 0.60 (45)	94.00 ± 0.21 (200)	94.55 ± 0.45 (166)	55.39 ± 1.29 (488)	92.58 ± 0.66 (69)
PCA	86.71 ± 0.36 (47)	94.91 ± 0.63 (41)	92.65 ± 0.75 (156)	93.42 ± 0.70 (94)	53.68 ± 1.81(24)	90.49 ± 0.90 (70)
LPP	80.37 ± 2.23 (45)	94.77 ± 0.39 (49)	91.01 ± 1.03 (198)	85.25 ± 1.02 (124)	52.74 ± 1.41(81)	91.76 ± 0.71 (67)
LSDA	81.20 ± 1.38 (41)	95.03 ± 0.49 (49)	92.19 ± 0.87 (128)	85.61 ± 1.26 (118)	56.02 ± 1.56 (482)	91.03 ± 1.48 (70)
RPCA-OM	91.21 ± 0.62 (41)	95.97 ± 0.54 (49)	94.27 ± 0.45 (114)	95.48 ± 0.29 (41)	64.62 ± 1.19 (219)	92.10 ± 0.77 (36)
RCDA	91.00 ± 0.64 (46)	96.72 ± 0.52 (33)	94.88 ± 0.93 (51)	94.72 ± 0.56 (36)	63.58 ± 0.66 (500)	92.48 ± 0.78 (60)
SDPCA	92.94 ± 0.87 (34)	97.29 ± 0.35 (35)	95.08 ± 0.91 (114)	96.91 ± 0.70 (106)	67.87 ± 0.72 (233)	98.51 ± 0.57 (31)
FAGPP	92.83 ± 0.71 (44)	96.97 ± 0.73 (45)	94.03 ± 0.97 (113)	95.22 ± 0.78 (140)	67.38 ± 1.19 (276)	98.60 ± 0.95 (70)

Note: Best results are in bold in the table.

From **Table 1**, it is obvious that our techniques outperform all the other

comparative algorithms in all the parameters they are evaluated on, except the standard deviations in a few cases. Thus, the proposed methods outperform all other methods in classification accuracy and optimal dimensions and in most cases consistency in performance across all the 21 random runs as can be noticed from the results. For instance, standard deviations of the proposed algorithms show consistent performance in all the 21 random runs. The IWPCA-1 method showed the most reliable performance in the MNIST dataset since smaller values are better for standard deviations. LPP had the most inconsistent performance in the MNIST dataset.

Figure 4 is an illustration of a plot of the average classification accuracies of the different algorithms on all six datasets whilst **Figure 5** demonstrates the classification accuracies across varying dimensions on the MNIST, USPS, ORL, COIL-20, FERET and UMIST datasets, respectively.

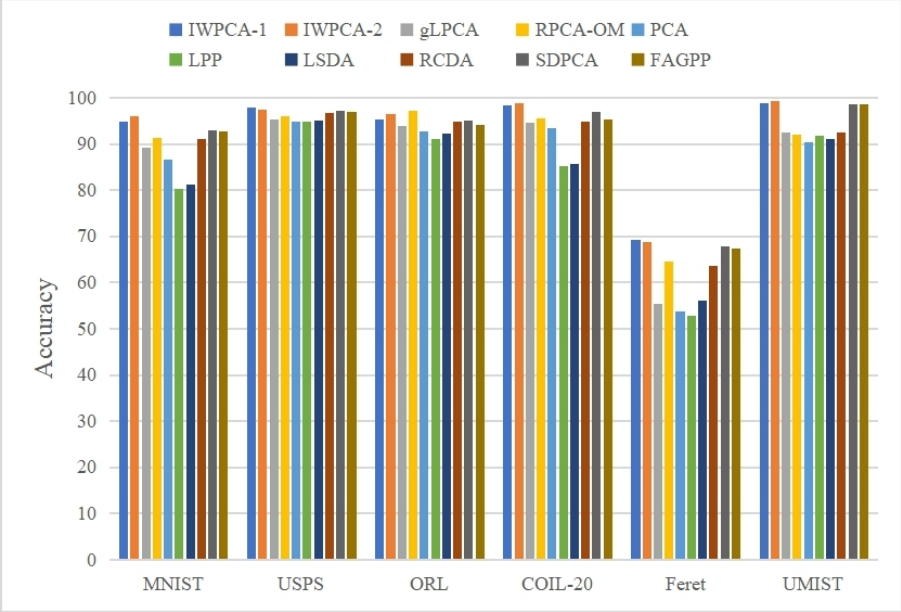
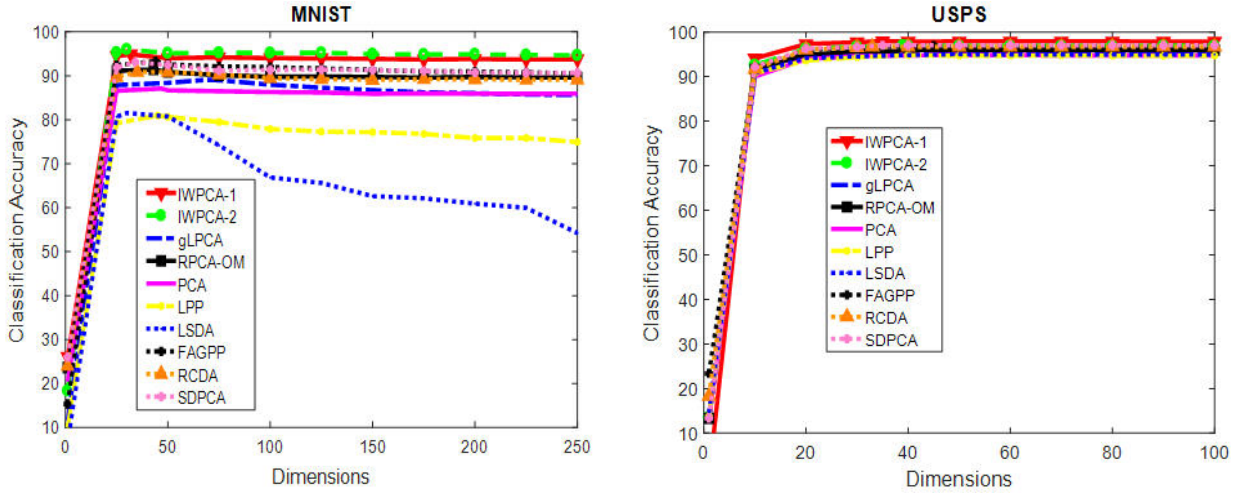


Figure 4. Average classification accuracies (%) of all methods on all six datasets.



(a) **(b)**

Figure 5. *Cont.*

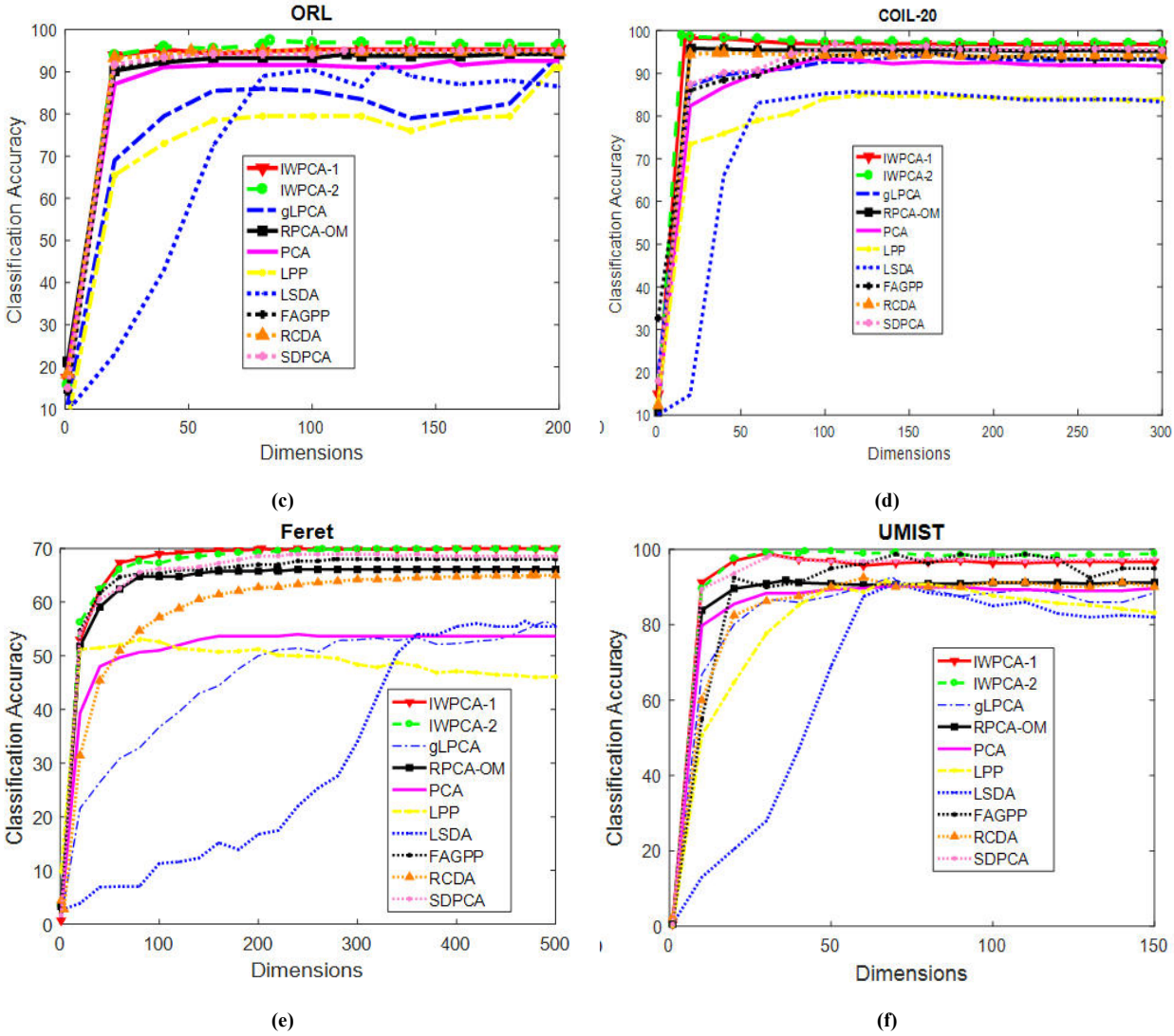


Figure 5. Classification accuracies (%) of all methods across different dimensions on the **(a)** MNIST, **(b)** USPS, **(c)** ORL, **(d)** COIL-20, **(e)** FERET and **(f)** UMIST datasets.

4.2.2. Experiments with noisy data

Although datasets generally consist of clean and noisy samples, we have introduced pepper and salt noise into the ORL (face) and COIL-20 (object) datasets to test how effective the proposed algorithm is at handling noise. We introduce 5% and 10% noise into these datasets and run each 21 times using each algorithm and recorded the classification accuracies. Sample images with noise are shown in **Table 2**, while image recognition accuracies are shown in **Table 3**.

Table 2. Sample images of ORL (face) and COIL-20 (object) datasets.

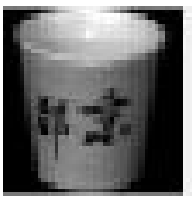
Dataset		ORL		COIL-20	
Original data	0%				

Table 2. *Cont.*

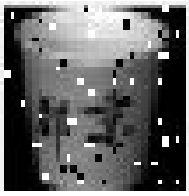
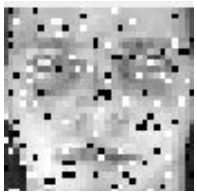
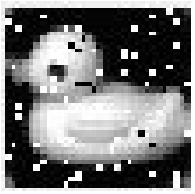
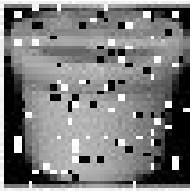
Dataset		ORL		COIL-20	
Corruption density	5%				
	10%				

Table 3. Average classification accuracies (%), standard deviations and optimal dimensions of all algorithms on corrupt ORL and COIL-20.

Dataset	**ORL**			**COIL-20**		
Corruption density	**0%**	**5%**	**10%**	**0%**	**5%**	**10%**
IWPCA-1	95.28 ± 0.13 (39)	93.93 ± 0.45 (42)	92.15 ± 0.78 (42)	98.34 ± 0.47 (17)	97.25 ± 0.68 (19)	95.36 ± 1.02 (25)
IWPCA-2	96.42 ± 0.50 (24)	95.31 ± 0.86 (27)	93.34 ± 0.97 (25)	98.84 ± 0.05 (15)	97.64 ± 0.13 (21)	96.05 ± 0.81 (23)
gLPCA	94.00 ± 0.21 (200)	92.08 ± 1.30 (200)	89.81 ± 1.58 (200)	94.55 ± 0.45 (166)	92.41 ± 1.13 (173)	89.66 ± 1.79 (181)
RPCA-OM	94.27 ± 0.45 (114)	92.37 ± 0.97 (123)	90.21 ± 1.09 (125)	95.48 ± 0.29 (41)	93.62 ± 0.65 (55)	91.35 ± 1.27 (71)
PCA	92.65 ± 0.75 (156)	90.44 ± 1.54 (162)	87.03 ± 1.76 (178)	93.42 ± 0.70 (94)	89.05 ± 1.45 (115)	86.21 ± 2.03 (126)
LPP	91.01 ± 1.03 (198)	89.03 ± 1.78 (200)	86.14 ± 1.92 (200)	85.25 ± 1.02 (124)	81.16 ± 1.71 (147)	77.09 ± 2.31 (151)
LSDA	92.19 ± 0.87 (128)	89.86 ± 1.91 (141)	86.13 ± 2.11 (156)	85.61 ± 1.26 (118)	80.11 ± 2.05 (122)	76.68 ± 2.61 (145)
RCDA	94.88 ± 0.93 (51)	92.93 ± 1.05 (59)	90.65 ± 1.82 (72)	94.72 ± 0.56 (36)	91.32 ± 0.98 (45)	88.80 ± 1.35 (53)
SDPCA	95.08 ± 0.91 (114)	93.19 ± 1.12 (120)	91.21 ± 1.45 (132)	96.91 ± 0.70 (106)	94.01 ± 0.89 (113)	93.72 ± 1.09 (128)
FAGPP	94.03 ± 0.97 (113)	92.05 ± 1.33 (118)	90.10 ± 1.68 (127)	95.22 ± 0.78 (140)	91.45 ± 1.17 (145)	89.42 ± 1.45 (160)

From **Table 3**, it can be noticed that for the ORL dataset, the cumulative decrements in classification accuracies when 5% and 10% noise were introduced for IWPCA-1, IWPCA 2, gLPCA, RPCA-OM, PCA, LPP, LSDA, RCDA, SDPCA and FAGPP are 3.13%, 3.08%, 4.19%, 4.09%, 5.62%, 4.87%, 6.06%, 4.23%, 3.87%, 3.93%, respectively. For the COIL-20 dataset, the cumulative reduction in accuracies for IWPCA-1, IWPCA-2, gLPCA, RPCA-OM, PCA, LPP, LSDA, RCDA, SDPCA after the noise is introduced are 2.98, 2.79, 4.89, 4.13, 7.21, 8.16, 8.93, 5.92, 3.29, 5.80, respectively.

The results show that although all the algorithms are sensitive to noise, the proposed methods are relatively less sensitive to noise. The results further reveal

that the proposed algorithms discover the intrinsic structure of the data in the midst of noise better than all the comparative methods. This is because it deals with noise more effectively than all the comparative methods, hence is less sensitive to noise or outliers.

4.2.3. Parameter sensitivity analysis

In this section, experiments are run using the IWPCA-2 algorithm with varying learning rate (γ) from 0.1 to 0.9 to determine if the proposed method is robust across these different learning rates. We use the MNIST, ORL, FERET, and, COIL-20 datasets to evaluate the sensitivity of the IWPCA-2 algorithm.

Figure 6 presents charts demonstrating how sensitive the IWPCA algorithm is to varying learning rates for the above datasets. From **Figure 6**, it can be observed that the performance of the IWPCA-2 algorithm is reliable for different learning rate values.

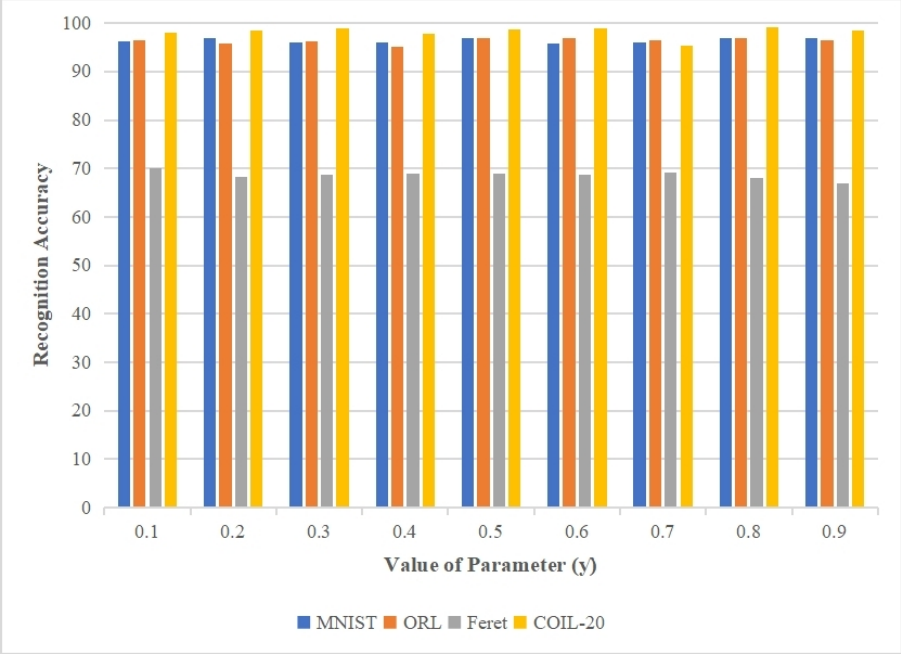


Figure 6. Sensitivity analysis of the IWPCA-2 algorithm for varying γ values.

4.2.4. Complexity analysis

All the algorithms were executed in MATLAB R2019b version 9.7 64-bit using a laptop with Intel (R) Core(TM) i7-1255U CPU @ 3.60 GHz, 16.00 GB memory and Windows 11 operating system.

The convergence of the proposed method depends on the iterative updating of the importance evaluation diagonal matrix D. For a training matrix of size m × n, it will take a computation time of O(m^3) to obtain the Eigen solution. Since the proposed algorithm is an eigenvalue problem, it will require O(m^3) to resolve it and obtain the projection vector w. But for the proposed algorithm to converge depends on the diagonal matrix D that evaluates the relevance of instances. Therefore, if it takes t and y iterations each for the "while" and "for" loops, respectively, in pursuing D to convergence, it will require $O\left(t(y(mn))\right)$. Consequently, the overall complexity of the algorithm becomes $O\left(m^3 + tymn\right)$.

The training and testing times for all 10 algorithms on all six datasets used for the

experiments are presented in **Table 4** as follows:

Table 4. Computation time of each algorithm in seconds for training (Tr) and testing (Te) on six datasets.

Dataset		IWPCA-1	IWPCA-2	RPCA-OM	PCA	RCDA	gLPCA	LPP	LSDA	FAGPP	SDPCA
MNIST	**Tr**	10.0	12.0	2.15	0.56	5.12	1.14	0.83	6.94×10^{-2}	2.10	9.25
	Te	5.36×10^{-2}	2.53×10^{-2}	2.61×10^{-2}	2.20×10^{-2}	1.72×10^{-2}	2.92×10^{-2}	1.89×10^{-2}	1.28×10^{-2}	5.32×10^{-2}	4.21×10^{-2}
USPS	**Tr**	15.01	20.61	2.17	0.35	2.12	3.21	1.25	1.95	3.50	11.81
	Te	5.25×10^{-2}	8.13×10^{-2}	3.89×10^{-2}	4.21×10^{-2}	2.0×10^{-2}	4.23×10^{-2}	3.12×10^{-3}	3.51×10^{-2}	4.0×10^{-2}	3.60×10^{-2}
ORL	**Tr**	8.33	13.98	4.53	0.56	4.87	4.11	0.42	0.27	6.91	8.12
	Te	5.21×10^{-2}	7.49×10^{-2}	8.01×10^{-2}	3.11×10^{-3}	4.77×10^{-2}	1.78×10^{-2}	2.24×10^{-3}	4.51×10^{-2}	5.26×10^{-2}	6.30×10^{-2}
COIL-20	**Tr**	4.45	5.62	1.29	0.49	3.45	0.55	0.85	6.38×10^{-1}	3.25	5.42
	Te	2.06×10^{-3}	1.60×10^{-2}	2.04×10^{-3}	1.95×10^{-2}	2.24×10^{-2}	1.74×10^{-2}	2.48×10^{-2}	1.44×10^{-2}	3.60×10^{-3}	1.01×10^{-2}
Feret	**Tr**	13.95	17.22	8.12	1.19	2.09×10^{-2}	1.93	1.40	1.41	12.15	3.71
	Te	5.78×10^{-2}	8.64×10^{-2}	3.96×10^{-2}	9.16×10^{-1}	3.45	1.13×10^{-3}	6.98×10^{-3}	5.50×10^{-3}	9.50×10^{-2}	5.51×10^{-2}
UMIST	**Tr**	9.54	11.40	8.12	1.09	2.25	1.93	1.40	1.41	11.31	7.21
	Te	3.92×10^{-2}	6.77×10^{-2}	3.96×10^{-2}	8.16×10^{-2}	2.09×10^{-2}	1.13×10^{-3}	6.98×10^{-3}	5.50×10^{-3}	5.3×10^{-2}	6.12×10^{-2}

Generally, the training times of the proposed algorithms are relatively higher than the comparative methods except SDPCA in the COIL-20 dataset and FAGPP in the UMIST dataset, respectively. This may be due to its iterative nature. The SDPCA obtained a higher training time than IWPCA-1 in the COIL-20 dataset. While FAGPP obtained a longer training time than IWPCA-1 in the UMIST dataset. The proposed method's training times are largely comparable to those of SDPCA and FAGPP although it obtained the highest recognition accuracies and reliable performances in all datasets.

5. Conclusion

We proposed in this study, an innovative generalized instance weighting PCA framework called IWPCA for dimensionality reduction. Unlike previous variations of PCA, our method embeds data in the projection space and applies instance weighting to find a solution to the missing observations and outlier challenges faced by traditional PCA, respectively. Distance and gradient measurements are used to build the instance importance evaluation diagonal matrix.

Comprehensive experimental evaluations on standard datasets with varying characteristics confirm how effective the proposed model is in classification tasks, as it consistently outperforms the other comparative methods. It achieves optimum performance in lower dimensions in all cases because it can scale down the effect of noise on the model. This additionally demonstrates that our algorithm is more outlier-tolerant and detects the underlying distribution among data than all the algorithms it was compared with.

We will extend this framework to graph embedding and low-rank representation with self-tuning parameters in the future.

Funding: This work received no external funding.

Institutional review board statement: Not applicable.

Informed consent statement: Not applicable.

Data availability statement: The datasets used for the experiments are available at https://doi.org/10.6084/m9.figshare.33130856.

Conflict of interest: The author declares no conflict of interest.

AI use statement: The author declares that no artificial intelligence (AI) tools were used in the preparation of this manuscript.

References

1. Hsiao JY, Chuang SJ, Chen PY. A hybrid face recognition system based on multiple facial features. International Journal of Computers and Applications. 2016; 38(1): 1–8. doi: 10.1080/1206212X.2016.1188553
2. Lin G, Tang N, Wang H. Locally principal component analysis based on L1-norm maximisation. IET Image Processing. 2015; 9(2): 91–96. doi: 10.1049/iet-ipr.2013.0851
3. Kong X, Hu C, Duan Z. Generalized Principal Component Analysis. In: Principal Component Analysis Networks and Algorithms. Springer; 2017. pp. 185–233. doi: 10.1007/978-981-10-2915-8_7
4. Ganaa ED, Abeo TA, Mehta S, et al. Incomplete-Data Oriented Dimension Reduction via Instance Factoring PCA Framework. In: Zhao Y, Barnes N, Chen B, et al. (editors). Image and Graphics. Springer; 2019. pp. 479–490. doi: 10.1007/978-3-030-34113-8_40
5. Cai D, He X, Zhou K, et al. Locality Sensitive Discriminant Analysis. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence; 6–12 January 2007; Hyderabad, India. pp. 1713–1726.
6. Li X, Chen M, Nie F, et al. Locality Adaptive Discriminant Analysis. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence; 19–25 August 2017; Melbourne, Australia. pp. 2201–2207. doi: 10.24963/ijcai.2017/306
7. Huang KK, Dai DQ, Ren CX. Regularized coplanar discriminant analysis for dimensionality reduction. Pattern Recognition. 2017; 62: 87–98. doi: 10.1016/j.patcog.2016.08.024
8. Li M, Luo X, Yang J, et al. Applying a Locally Linear Embedding Algorithm for Feature Extraction and Visualization of MI-EEG. Journal of Sensors. 2016; 2016: 1–9. doi: 10.1155/2016/7481946
9. Feng G, Hu D, Zhou Z. A Direct Locality Preserving Projections (DLPP) Algorithm for Image Recognition. Neural Processing Letters. 2008; 27(3): 247–255. doi: 10.1007/s11063-008-9073-1
10. Wang J, Wu Y, Li B, et al. Fast anchor graph preserving projections. Pattern Recognition. 2024; 146: 109996. doi: 10.1016/j.patcog.2023.109996
11. Zhang F, Wang H, Qin W, et al. Generalized nonconvex regularization for tensor RPCA and its applications in visual inpainting. Applied Intelligence. 2023; 53(20): 23124–23146. doi: 10.1007/s10489-023-04744-9
12. Xu H, Caramanis C, Sanghavi S. Robust PCA via Outlier Pursuit. IEEE Transactions on Information Theory. 2012; 58(5): 3047–3064. doi: 10.1109/TIT.2011.2173156
13. Netrapalli P, Niranjan UN, Sanghavi S, et al. Non-convex robust pca. In: Proceedings of the Advances in Neural Information Processing Systems 27; 8–13 December 2014; Montreal, QC, Canada.
14. Jamil NW, Sharif S. Outlier Map of Classical and Robust Principal Component Analysis. Applied Mathematics and Computational Intelligence (AMCI). 2025; 14(3): 37–46. doi: 10.58915/amci.v14i3.2071
15. Ouermi TAJ, Li J, Johnson CR. A Fast Iterative Robust Principal Component Analysis Method. arXiv preprint. 2025; arXiv:2506.16013. doi: 10.48550/arXiv.2506.16013
16. Nie F, Yuan J, Huang H. Optimal mean robust principal component analysis. International Conference on Machine Learning. 2014; 32(2): 1062–1070.
17. Jana T, Raghav N, Majumdar A, et al. Exact Rotation Invariant Robust PCA. In: Proceedings of the 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP); 6–11 April 2025; Hyderabad, India. pp. 1–5. doi: 10.1109/ICASSP49660.2025.10889970
18. Hu Z, Pan G, Wang Y, et al. Sparse Principal Component Analysis via Rotation and Truncation. In: Naik GR (editor). Advances in Principal Component Analysis. Springer; 2018. pp. 1–18. doi: 10.1007/978-981-10-6704-4_1
19. Jenatton R, Obozinski G, Bach F. Structured sparse principal component analysis. Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. 2010; 9: 366–373.
20. Jiang B, Ding C, Luo B, et al. Graph-Laplacian PCA: Closed-Form Solution and Robustness. In: Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition; 23–28 June 2013; Portland, OR, USA. pp. 3492–3498. doi: 10.1109/CVPR.2013.448
21. Kumar N, Singh S, Kumar A. Random permutation principal component analysis for cancelable biometric

recognition. Applied Intelligence. 2018; 48(9): 2824–2836. doi: 10.1007/s10489-017-1117-7
22. Zhao Q, Meng D, Xu Z, et al. Robust principal component analysis with complex noise. Proceedings of the 31st International Conference on Machine Learning. 2014; 32(2): 55–63.
23. Tabaghi P, Khanzadeh M, Wang Y, et al. Principal Component Analysis in Space Forms. IEEE Transactions on Signal Processing. 2024; 72: 4428–4443. doi: 10.1109/TSP.2024.3457529
24. Li X, Li P, Zhang H, et al. Pivotal-Aware Principal Component Analysis. IEEE Transactions on Neural Networks and Learning Systems. 2024; 35(9): 12201–12210. doi: 10.1109/TNNLS.2023.3252602
25. Mi JX, Zhu Q, Lu J. Principal component analysis based on block-norm minimization. Applied Intelligence. 2019; 49(6): 2169–2177. doi: 10.1007/s10489-018-1382-0
26. Hubert M, Rousseeuw PJ, Vanden Branden K. ROBPCA: A New Approach to Robust Principal Component Analysis. Technometrics. 2005; 47(1): 64–79. doi: 10.1198/004017004000000563
27. Xu H, Caramanis C, Mannor S. Outlier-Robust PCA: The High-Dimensional Case. IEEE Transactions on Information Theory. 2013; 59(1): 546–572. doi: 10.1109/TIT.2012.2212415
28. Leyder S, Raymaekers J, Verdonck T. Generalized spherical principal component analysis. Statistics and Computing. 2024; 34(3): 104. doi: 10.1007/s11222-024-10413-9
29. Wang S, Nie F, Wang Z, et al. Max–Min Robust Principal Component Analysis. Neurocomputing. 2023; 521: 89–98. doi: 10.1016/j.neucom.2022.11.092
30. Zhan J, Vaswani N. Robust PCA With Partial Subspace Knowledge. IEEE Transactions on Signal Processing. 2015; 63(13): 3332–3347. doi: 10.1109/TSP.2015.2421485
31. Zhou P, Feng J. Outlier-Robust Tensor PCA. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR); 21–26 July 2017; Honolulu, HI, USA. pp. 3938–3946. doi: 10.1109/CVPR.2017.419
32. Higuchi I, Eguchi S. Robust principal component analysis with adaptive selection for tuning parameters. Journal of Machine Learning Research. 2004; 5: 453–471.
33. Candès EJ, Li X, Ma Y, et al. Robust principal component analysis? Journal of the ACM. 2011; 58(3): 1–37. doi: 10.1145/1970392.1970395
34. Hubert M, Rousseeuw P, Verdonck T. Robust PCA for skewed data and its outlier map. Computational Statistics & Data Analysis. 2009; 53(6): 2264–2274. doi: 10.1016/j.csda.2008.05.027
35. Xu X, Gao R, Qing Y, et al. Hyperspectral image mixed noise removal via tensor robust principal component analysis with tensor-ring decomposition. International Journal of Remote Sensing. 2023; 44(5): 1556–1578. doi: 10.1080/01431161.2023.2187720
36. Zhou Q, Gao Q, Wang Q, et al. Sparse discriminant PCA based on contrastive learning and class-specificity distribution. Neural Networks. 2023; 167: 775–786. doi: 10.1016/j.neunet.2023.08.061
37. Shen H, Huang JZ. Sparse principal component analysis via regularized low rank matrix approximation. Journal of Multivariate Analysis. 2008; 99(6): 1015–1034. doi: 10.1016/j.jmva.2007.06.007
38. Chen X, Sun H. $L_{2,1}$ -norm-based sparse principle component analysis with trace norm regularised term. IET Image Processing. 2019; 13(6): 910–922. doi: 10.1049/iet-ipr.2018.5433
39. Bertsimas D, Cory-Wright R, Pauphilet J. Solving large-scale sparse pca to certifiable (near) optimality. Journal of Machine Learning Research. 2022; 23(13): 1–35.
40. Bertsimas D, Kitane DL. Sparse pca: a geometric approach. Journal of Machine Learning Research. 2023; 24(32): 1–33.
41. Yi S, Lai Z, He Z, et al. Joint sparse principal component analysis. Pattern Recognition. 2017; 61: 524–536. doi: 10.1016/j.patcog.2016.08.025
42. Van Deun K, Thorrez L, Coccia M, et al. Weighted sparse principal component analysis. Chemometrics and Intelligent Laboratory Systems. 2019; 195: 103875. doi: 10.1016/j.chemolab.2019.103875
43. Del Pia A. Sparse PCA on fixed-rank matrices. Mathematical Programming. 2023; 198(1): 139–157. doi: 10.1007/s10107-022-01769-9
44. Mackey L. Deflation methods for sparse pca. In: Proceedings of the Advances in Neural Information Processing Systems 21; 12–13 December 2008; Vancouver, BC, Canada.
45. Khanna R, Ghosh J, Poldrack R, et al. A Deflation Method for Structured Probabilistic PCA. In: Proceedings of the 2017 SIAM International Conference on Data Mining; 27–29 April 2017; Houston, TX, USA. pp. 534–542. doi: 10.1137/1.9781611974973.60
46. Ganaa ED, Abeo TA, Wang L, et al. Robust deflated principal component analysis via multiple instance factorings for dimension reduction in remote sensing images. Journal of Applied Remote Sensing. 2020; 14(3): 1. doi:

10.1117/1.JRS.14.032608
47. Khanna R, Ghosh J, Poldrack R, et al. A Deflation Method for Probabilistic PCA. In: Proceedings of the 29th Annual Conference on Neural Information Processing Systems; 11–12 December 2015; Montreal, QC, Canada.
48. Hassani S, Hanafi M, Qannari EM, et al. Deflation strategies for multi-block principal component analysis revisited. Chemometrics and Intelligent Laboratory Systems. 2013; 120: 154–168. doi: 10.1016/j.chemolab.2012.08.011
49. Xu C, Yang M, Zhang J. Fast deflation sparse principal component analysis via subspace projections. Journal of Statistical Computation and Simulation. 2020; 90(8): 1399–1412. doi: 10.1080/00949655.2020.1728761
50. Shi Q, Lu H, Cheung YM. Rank-One Matrix Completion with Automatic Rank Estimation via L1-Norm Regularization. IEEE Transactions on Neural Networks and Learning Systems. 2018; 29(10): 4744–4757. doi: 10.1109/TNNLS.2017.2766160
51. Khishe M, Azar OP, Hashemzadeh E. Variable-length CNNs evolved by digitized chimp optimization algorithm for deep learning applications. Multimedia Tools and Applications. 2024; 83(1): 2589–2607. doi: 10.1007/s11042-023-15411-z
52. Kosarirad H, Ghasempour Nejati M, Saffari A, et al. Feature Selection and Training Multilayer Perceptron Neural Networks Using Grasshopper Optimization Algorithm for Design Optimal Classifier of Big Data Sonar. Journal of Sensors. 2022; 2022: 1–14. doi: 10.1155/2022/9620555
53. Dash CSK, Behera AK, Dehuri S, et al. An outliers detection and elimination framework in classification task of data mining. Decision Analytics Journal. 2023; 6: 100164. doi: 10.1016/j.dajour.2023.100164
54. Ansari S, Nassif AB, Mahmoud S, et al. Impact of Outliers on Regression and Classification Models: An Empirical Analysis. In: Proceedings of the 2024 17th International Conference on Development in eSystem Engineering (DeSE); 6–8 November 2024; Khorfakkan, United Arab Emirates. pp. 211–218. doi: 10.1109/DeSE63988.2024.10912020