Spectrum of Engineering Sciences
ISSN (e) 3007-3138 (p) 3007-312X Volume 3, Issue 4, 2025

UNDERSTANDING PRINCIPAL COMPONENT ANALYSIS (PCA): A LINEAR ALGEBRA APPROACH TO DIMENSIONALITY REDUCTION

Sohail Ahmed Memon*[1], Imtiaz Ahmed[2], Shoaibullah[3]

*[1,2,3]*Department of Mathematics, Shah Abdul Latif University, Khairpur Mirs*

*[1]suhail.memon@salu.edu.pk, [2]sharimtiaz2014@gmail.com, [3]shoaibpk00@gmail.com

DOI: https://doi.org/10.5281/zenodo.15771259

Keywords
Principal Component Analysis (PCA); Dimensionality Reduction; Multivariate Data Analysis; Machine Learning Pre-processing; Classification; Multi-Classification

Article History
Received on O5 February 2025
Accepted on 21 April 2025
Published on 29 April 2025

Copyright @Author
Corresponding Author: *
Sohail Ahmed Memon

Abstract

Principal Component Analysis (PCA) is one of the most widely used techniques for dimensionality reduction in data analysis and machine learning. This work offers a mathematical based introduction to PCA, presenting its interpretation through the perspective of linear algebra. We begin by giving our view about the motivation for dimensionality reduction and by introducing the foundational concepts such as vectors, matrices, covariance, and eigen decomposition. We then present our work starting from the PCA algorithm step by step for projecting it onto a lower dimensional subspace reduced from the centred data. An example based on two-dimensional Seeds dataset is demonstrated to explain the entire process, which is supported by implementing machine learning model (LightGMB classifier) and staging visualizations. For this model, we achieved an improved accuracy score (ROC AUC score) after applying PCA and have discussed the comparison of classification performances. We further explore practical applications of PCA in image compression, noise reduction and machine learning. Finally, we discuss the strengths and limitations of PCA, highlighting when it is appropriate and when more complex techniques may be essential. The work is produced for the machine learning practitioners with a basic understanding of linear algebra and programming.

INTRODUCTION

In existing data analysis and machine learning, datasets often comprise a large number of features which comprise hundreds or thousands per observation. The high dimensional data being informative in most cases, poses computational and analytical challenges. Such huge data often cause overfitting in predictive models, enhanced training time, and complications in visualizing and proper interpreting the structure of data. Such situation is often related to the curse of dimensionality [1]. As the dimensionality increases, the volume of the feature space grows exponentially, where data points become sparse and conclusions become statistically less reliable. A powerful statistical tool properly known as Principal Component Analysis (PCA) handles these challenges by transforming high dimensional data into lower dimensional data. The PCA approach preserves the structure that contributes most to its variance. It employs orthogonal directions which are called principal components, along which data varies the most [2]. The implementation of PCA in this study achieves dimensionality reduction, retaining the most meaningful variation in the data. In early 1900's, Karl Pearson (1901) introduced a method of finding the best-fitting linear subspace for multivariate data [3], a type dimensionality reduction. Later, it was adopted more formally as PCA technique by

Hotelling (1993) in a study based on spatial genetic structure [4]. The technique of PCA has been used since then and has become a cornerstone in both supervised and unsupervised learning. The PCA is commonly used in image compression, gene expression analysis, face recognition, and more [5].

There are several undeniable reasons to employ PCA in handling high-dimensional data. In visualization, data with many features can be visualized more effectively using the top few principal components. PCA can help eliminate noise by discarding components with small variances that often correspond to measurement errors. It is often used in machine learning pipelines to reduce dimensionality prior to model training for improved generalization [6]. By keeping the top components using this approach, a minimum storage can be used and a significant computational efficiency can be achieved.

The technique of PCA works as linear method, it captures linear relationships in the data where nonlinear relationships are prevailing. The examples of these techniques, which are mostly preferred, include Kernel PCA [7], t-distributed Stochastic Neighbour Embedding (t-SNE) [8], and Autoencoders [9].

This article presents a structured, linear algebra-based introduction to PCA with a mathematical intuition behind this technique. To demonstrate the application of PCA, we have utilized seeds dataset. It is well known dataset in pattern recognition and clustering. The dataset consists of 210 samples of three different varieties (Kama, Rose, Canadian) of wheat. Each sample is described using seven numerical features derived from geometrical properties of wheat kernels, such as area, perimeter, compactness, kernel length, width, asymmetry coefficient, and groove length. The technique of PCA is employed to project the seven-dimensional data into lower dimensions, allowing us to explore how well-separated the wheat varieties are in the principal component space.

Readers will gain insight into the geometric interpretation of PCA, learn to compute it step-by-step, and explore real-world applications through illustrative examples. By the end, readers should be equipped to apply PCA thoughtfully, understand its limitations, and recognize when more advanced nonlinear methods are necessary.

1. Literature Review

Principal Component Analysis (PCA) is a vastly used as a dimensionality reduction tool in data science, machine learning and signal processing. Through this tool, a suitable set of features can be selected to obtain improved accuracy of a predictive model. Its importance in diverse applications have made PCA a subject of extensive theoretical and practical study.

The foundation of this technique was first laid by karl Pearson in 1901 [3] to minimize dimensionality by projecting data onto directions of maximum variance. After that, it was implemented by Hotelling [4] for multivariate statistical analysis where interpretations were performed through eigen decomposition of the covariance matrix. The curse of dimensionality indicated by Bellman [1] refers to various occurrences that arise when working with high-dimensional spaces, like data sparsity and overfitting in machine learning models. In such situations, the dimensionality reduction techniques such as PCA play an important role as powerful tools for the improved generalization, reduction of computational resources, and enabling data visualizations. PCA has is a mathematical technique based on linear algebra, particularly the eigen decompositions of the covariance matrix or the singular value decomposition (SVD) of data matrix. The eigenvectors of the covariance matrix indicate the directions of greatest variance and eigenvalues represent the magnitude of variance along these directions [10]. Such understandings not only make PCA mathematically sophisticated but also provide strong geometric intuition. PCA obtains a low-dimensional affine subspace that best approximates the data in the least-squares sense for a centred data [11].

PCA has applications in bioinformatics which made possible to visualize high-dimensional gene expression data [12]. It has also been helpful in image compression to reduce the storage of high-dimensional image data by retaining the most important components [13]. The applications that are developed for finance employ PCA to set out model interest rate term structures and to reduce risk exposure by identifying dominant modes in asset returns [14]. Moreover, PCA has been helpful for noise reduction and visualization in fault detection, medical diagnostics, remote sensing, and speech and handwritten recognition.

Most importantly, PCA is often used for feature pre-processing steps in machine leaning pipelines. It helps to remove multicollinearity and reduce training time [15].

Despite the importance and usefulness of PCA, it has limitations. It only obtains linear correlations and is sensitive to data scaling. To deal with such limitations, the researchers have proposed several other variants. Among these variants, the Kernel PCA [7] maps data to a higher-dimensional feature space using kernel functions before applying PCA. Another variant Robust PCA [16], decomposes data into low-rank and sparse matrices, which make such decomposition of data stronger against outliers. A variant of PCA called Sparse PCA [17] handles sparsity to make principal components more useful and interpretable. There are other nonlinear methods such as Autoencoders and t-SNE are as alternatives, but the PCA remains dominant due to its solid theoretical foundation, interpretability, and simplicity.

The recent work based on PCA include gene expression data from COVID-19 patients. It was used in pre-processing to visualize patient clustering and to reduce noise for deep leaning models [18]. PCA technique was to compress hyperspectral imaging data, where it enabled more efficient classification using convolutional neural networks, such utilization reduced overfitting [19]. The implementation of PCA has been significant in biomedical domain, it demonstrated in pre-processing EEG and ECG signals to extract signals without noise. This utility of PCA achieved improved accuracy in health condition classification based on the long short-term memory (LSTM) models where compressed representations were fed [20]. Also, the technique of PCA has been utilized for unsupervised clustering of MRI brain scans, contributing to early detection of neurodegenerative diseases [21].

The technique of PCA has leveraged its functionality in the field of cybersecurity. It has been used to isolate anomalies in network traffic data by improving intrusion detection performance in a low computation cost [22].

Moreover, the importance of PCA in exploratory data analysis has been very helpful where it has served as a gateway technique to understanding more complex dimensionality reduction methods. It has been applied various datasets for learning purposes. These datasets included Iris, Breast Cancer and other datasets available at Kaggle. This study focuses on PCA from a linear algebraic viewpoint, we use Seeds dataset as a case study to visualize the effect of dimensionality reduction.

2. Mathematical Foundations

In this section, the important mathematical tools are presented that form the backbone of PCA. It relies on fundamental linear algebra concepts such as vectors, projections, matrices, covariance, and eigen decomposition. This section provides a concise review of these concepts to build intuition of working of PCA.

First of all, we illustrate the vectors and linear transformations. A vector represents a point or direction in space, commonly express as column of numbers. A linear transformation maps vectors from one space to another using a matrix.

In PCA, the data points are treated as vectors $\mathbb{R}^d$ space and seek transformations that align them along principal axes.

The dot product $\mathbf{u}$ and $\mathbf{v}$ is given by:

$$\mathbf{u} \cdot \mathbf{v} = \|\mathbf{u}\|\|\mathbf{v}\| cos\theta \qquad (1)$$

The expression given in equation (1) quantifies the directional similarity between two vectors and is important in projections and principal component analysis.

The projection of a vector $\mathbf{x}$ onto another vector $\mathbf{w}$, which is given as follows:

$$\mathrm{proj}_{\mathbf{w}}(\mathbf{x}) = \frac{\mathbf{x} \cdot \mathbf{w}}{\|\mathbf{w}\|^2}\mathbf{w} \qquad (2)$$

PCA projects data points onto directions (principal components) that maximize variance [10].

Given a dataset with n samples and d features, we arrange the data in an $n \times d$ matrix X, where row is a data point. It must be made sure before computing the covariance matrix, the data must be centred and each feature has zero mean.

The covariance matrix is defined as:

$$\Sigma = \frac{1}{n-1} X^{\mathrm{T}} X \qquad (3)$$

Basically, the covariance matrix given in equation (2) is square matrix, a type of symmetric table that shows how the feature in the data relate to each other. The values on the diagonal indicate how much each individual feature varies by itself. This is called the variance of that feature. The values off the diagonal inform how two different features change together and this is called the covariance between those two features. Simply, the matrix helps to understand not only how spread out each feature is, but also whether some features increase or decrease [5].

Following we show how eigenvalues and eigenvectors contribute to PCA by illustrating the fundamental concept.

Let $A \in \mathbb{R}^{d \times d}$ be a square matrix. A non-zero vector $\mathbf{v}$ is called an eigenvector of A if:

$$A\mathbf{v} = \lambda\mathbf{v} \qquad (4)$$

where λ is the corresponding eigenvalue.

The eigenvalues and eigenvectors of covariance matrix are calculated in PCA. The sample equation is given for covariance matrix calculation is given in equation (3) and equation for eigenvalues and eigenvectors is given in equation (4). It must be noted that the eigenvalues measure the variance captured along the principal directions represented by eigenvectors. The first principal component corresponds to the eigenvector with the largest eigenvalue and then a set of principal components is built through sorting eigenvectors by descending eigenvalues. Such decomposition lays out the strategy to reduce the dimensionality of the data by selecting only top k eigenvectors [23], [24].

3. General Implementation of PCA

This section presents the step by step mathematical process of principal component analysis. PCA works by transforming the original dataset into a new coordinate system where the axes, generally called principal components, represent directions of maximum variance. The overall process is presented in five fundamental steps based on linear algebra.

3.1 Centring the Data

Centering the data is a first step in PCA, the mean of each feature is subtracted from actual features to obtain zero mean. This step is crucial for PCA for making the first principal component correspond to the direction of maximum variance. The mathematical for of this step is given in equation (5), given as follows:

$$\tilde{X} = X - \mu \qquad (5)$$

where μ is the mean vector computed across all rows (samples).

4.2 Computing the Covariance Matrix

In the second step, the covariance matrix is computing using equation (3). The covariance matrix obtains the variance of each feature and the covariance between each pair of features. The matrix contained in covariance matrix is symmetric and yields insights into the structure of the data.

4.3 Performing Eigen Decomposition

Third step performs the eigen decomposition of the covariance matrix using equation (4) where A is covariance matrix. This step provides a set of eigenvectors (principal directions) and eigenvalues (variance explained along each direction).

4.4 Selecting top-k Eigenvectors

The fourth step is about selecting top-k eigenvectors which correspond to the largest eigenvalues (directions of highest variance). This process reduces the dimensionality by sorting eigenvalues in descending order and selecting the corresponding k eigenvectors.

4.5 Projecting the Data onto the New Subspace

Finally, the step five performs the projection of data onto a new subspace. Once there are the top-k eigenvectors, the original data is projected not the new subspace. The process presents the reduced representation of the data. The mathematical form of this step is given in below:

$$Z = \tilde{X}.W_k \qquad (6)$$

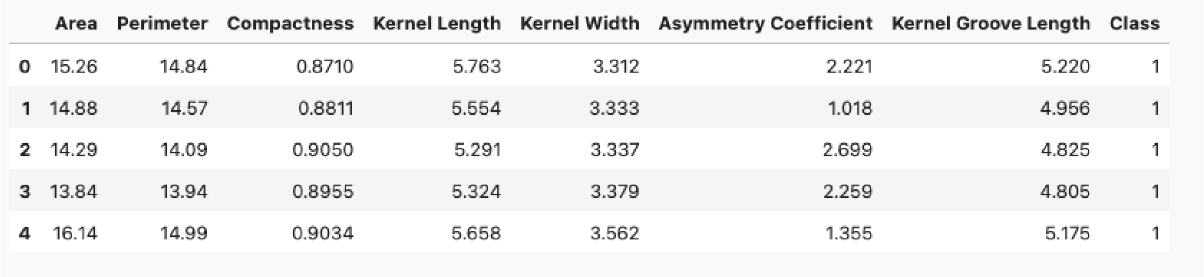
Where Z is the projected data and W_k is the matrix of top-k eigenvectors.

4. PCA Implementation on the Seeds Dataset

To present the practical implementation of Principal Component Analysis (PCA), it is employed to a real-world classification dataset. This is Seeds Dataset which is taken from the Kaggle datasets and is available at:

https://www.kaggle.com/datasets/rwzhang/seeds-dataset.

The dataset contains seven numerical features extracted from geometric properties of wheat kernels, and each instance belongs to one of three wheat varieties. These varieties are comprised of Kama, Rosa, and Canadian. The dataset contains 210 samples and seven features, which were derived from X-ray images, of wheat kernels. The column names or features in the dataset are Area, Perimeter, Compactness, Kernel Length, Kernel Width, Asymmetry Coefficient, and Length Kernel Groove. The target variable is Class which indicates the wheat variety (1, 2 and 3). A snap shot of the head of dataset is shown in Figure 1.



	Area	Perimeter	Compactness	Kernel Length	Kernel Width	Asymmetry Coefficient	Kernel Groove Length	Class
0	15.26	14.84	0.8710	5.763	3.312	2.221	5.220	1
1	14.88	14.57	0.8811	5.554	3.333	1.018	4.956	1
2	14.29	14.09	0.9050	5.291	3.337	2.699	4.825	1
3	13.84	13.94	0.8955	5.324	3.379	2.259	4.805	1
4	16.14	14.99	0.9034	5.658	3.562	1.355	5.175	1

Figure 1. The first five rows of the Seeds dataset.

It is a multiclass classification problem based on moderately small number of features and is chosen in this study to demonstrate the application of PCA. However, the PCA can be implemented on large size datasets to reduce the dimensionality.

The dataset does not contain any missing or categorical values. Only the standardization was applied using StandardScalar utility to have zero mean and unit variance. This standardization is necessary for PCA due to its sensitivity to scale of the input features. The dataset was split into train and test sets into the slice of 0.80 and 0.20 respectively. The LightGBM (LGBM) classifier machine learning model has been applied to evaluate classification performance before and after applying PCA.

The PCA was applied to reduce the original 7-dimensional datasets into 4 principal components. The transformed data after applying PCA showed the three wheat classes are separated in the two-dimensional subspace.

We have used ROC AUC score (macro-average) as the evaluation metric for multiclass classification. The ROC AUC score before applying PCA was 0.9754, whereas after dimensionality reduction using PCA, it improved slightly to 0.9788. The plot shown in Figure 2 clearly displays ROC curves for multiclass classification task performed using LGBM classifier, both before and after applying PCA to the feature space.

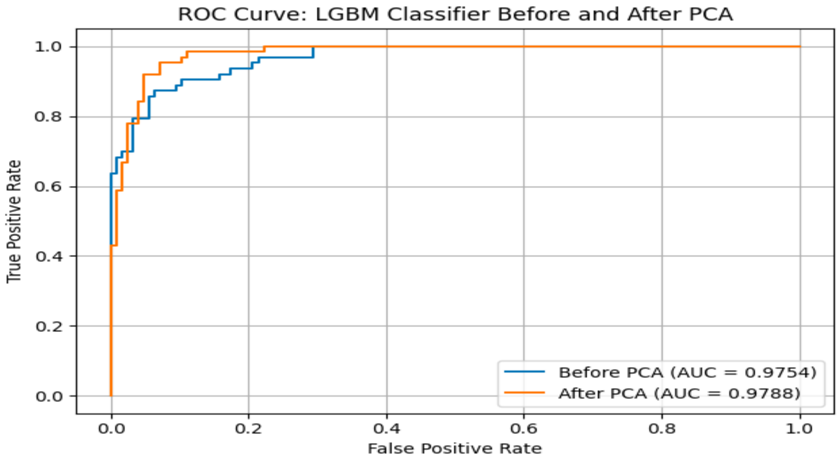


Figure 2. ROC Curves for the LGBM Classifier before and after applying PCA.

The comparison of ROC curves has been shown in Figure 2 for a classification performance on the Seeds dataset using the original feature space. A clear difference can be observed after PCA, with AUC increasing from 0.9754 to 0.9788, indicating effective dimensionality reduction without loss of discriminative power.

5. Conclusions

In this study, we have presented the theoretical foundations and practical applications of Principal Component Analysis (PCA) in perspective of linear algebra. The technique of PCA is vastly utilized for dimensionality reduction that transforms high-dimensional data into a lower-dimensional subspace while preserving the directions of maximum variance. PCA enables simplification of datasets while retaining their essential structure. This process involves centring the data, computing the covariance matrix, performing eigen decomposition, and projecting onto the top principal components.

To better understand the implementation of PCA, we conducted a worked example using the Seeds dataset, a real-world classification problem for the identification of wheat varieties based on geometrical features. A comparison was presented for classification performance using LightGBM classifier which was trained on the original features and on the transformed data using PCA. The ROC AUC scores before and after PCA were 0.9745 and 0.9788. We achieved a small improved accuracy score while reducing the feature space.

Basically, the PCA is based on linear relationships for and may not perform optimally for highly nonlinear datasets. Its simplicity, efficiency and interpretability make it a foundational technique in data machine learning and data science. For more complex data or higher dimensional data, other advanced techniques like Kernel PCA or nonlinear dimensionality reduction methods such as t-SNE or UMAP can be benefited. The transformed data through these advanced techniques can be employed to stronger machine learning or deep learning models to the get higher accuracy.

This work presents PCA and its relevance as a conceptual framework and a practical tool for the transformation high-dimensional data into meaningful lower-dimensional representations. It serves as a bridge between mathematical theory and machine learning practice, highlighting the power of linear algebra in solving real-world data problems.

REFERENCES

[1] R. Bellman and R. Kalaba, "On adaptive control processes," *IRE Trans. Autom. Control*, vol. 4, no. 2, pp. 1–9, 1959.

[2] "Principal Component Analysis for Special Types of Data," in *Principal Component Analysis*, in Springer Series in Statistics. , New York: Springer-Verlag, 2002, pp. 338–372. doi: 10.1007/0-387-22440-8_13.

[3] K. Pearson, "LIII. *On lines and planes of closest fit to systems of points in space*," *Lond. Edinb. Dublin Philos. Mag. J. Sci.*, vol. 2, no. 11, pp. 559–572, Nov. 1901, doi: 10.1080/14786440109462720.

[4] H. Hotelling, "Analysis of a complex of statistical variables into principal components.," *J. Educ. Psychol.*, vol. 24, no. 6, p. 417, 1933.

[5] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philos. Trans. R. Soc. Math. Phys. Eng. Sci.*, vol. 374, no. 2065, p. 20150202, Apr. 2016, doi: 10.1098/rsta.2015.0202.

[6] T. Hastie, R. Tibshirani, and J. Friedman, "The elements of statistical learning." Citeseer, 2009.

[7] B. Schölkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, 1998.

[8] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE.," *J. Mach. Learn. Res.*, vol. 9, no. 11, 2008.

[9] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[10] G. Strang, *Introduction to linear algebra*. SIAM, 2022.

[11] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*, vol. 4. Springer, 2006.

[12] M. Ringnér, "What is principal component analysis?," *Nat. Biotechnol.*, vol. 26, no. 3, pp. 303–304, 2008.

[13] M. Turk and A. Pentland, “Eigenfaces for recognition,” *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[14] C. Alexander, *Market models: A guide to financial data analysis*. University of Sussex, 2001.

[15] J. Chen and W. K. Jenkins, “Facial recognition with PCA and machine learning methods,” in *2017 IEEE 60th international Midwest symposium on circuits and systems (MWSCAS)*, IEEE, 2017, pp. 973–976.

[16] E. J. Candès, X. Li, Y. Ma, and J. Wright, “Robust principal component analysis?,” *J. ACM*, vol. 58, no. 3, pp. 1–37, May 2011, doi: 10.1145/1970392.1970395.

[17] H. Zou, T. Hastie, and R. Tibshirani, “Sparse Principal Component Analysis,” *J. Comput. Graph. Stat.*, vol. 15, no. 2, pp. 265–286, Jun. 2006, doi: 10.1198/106186006X113430.

[18] K. Fujisawa1E, M. Shimo, Y. H. Taguchi, S. Ikematsu, and R. Miyata5E, “PCA-based unsupervised feature extraction for gene expression analysis of COVID-19 patients”.

[19] H. Hasan, H. Z. Shafri, and M. Habshi, “A comparison between support vector machine (SVM) and convolutional neural network (CNN) models for hyperspectral image classification,” in *IOP Conference Series: Earth and Environmental Science*, IOP Publishing, 2019, p. 012035.

[20] J.-N. Lee and K.-C. Kwak, “ECG-based biometrics using a deep network based on independent component analysis,” *IEEE Access*, vol. 10, pp. 12913–12926, 2022.

[21] D. G. O. Olle, J. Z. Bisse, and G. A. Alo’o, “Application and comparison of K-means and PCA based segmentation models for Alzheimer disease detection using MRI,” *Discov. Artif. Intell.*, vol. 4, no. 1, p. 11, 2024.

[22] F. Nabi and X. Zhou, “Enhancing intrusion detection systems through dimensionality reduction: A comparative study of machine learning techniques for cyber security,” *Cyber Secur. Appl.*, vol. 2, p. 100033, 2024.

[23] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*, vol. 4. Springer, 2006.

[24] T. Hastie, R. Tibshirani, and J. Friedman, “The elements of statistical learning.” Citeseer, 2009.