RECENT APPLICATIONS OF PCA AND SVD

Thesis submitted
in Partial Fulfillment of the Requirement for the
Degree of

MASTER OF SCIENCE

IN

APPLIED MATHEMATICS

by

Aditi Ahuja
(23/MSCMAT/58)

Under the supervision of
Mr. Jamkhongam Touthang
Department of Applied Mathematics
Delhi Technological University

DEPARTMENT OF APPLIED MATHEMATICS
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur , Main Bawana Road, Delhi – 110042 , India

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur , Main Bawana Road, Delhi – 110042

CANDIDATE'S DECLARATION

I, **Aditi**, Roll No. 23/ MSCMAT/58, hereby certify that the work which is being presented entitled *"RECENT APPLICATIONS OF PCA AND SVD"*,in partial fulfillment of the requirements for the degree of Master of Science, submitted in Department of Applied Mathematics, Delhi Technological University is an authentic record of our own work carried out during the period from August 2024 to May 2025 under the supervision of Mr. Jamkhongam Touthang.
The matter presented in the thesis has not been submitted by us for the award of any other degrees of this or any other Institute.

Candidate's Signature

This is to certify that the students have incorporated all the correction suggested by the examiners in the thesis and the statement made by the candidate to the best of our knowledge.

Signature of Supervisior **Signature of External Examiner**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur , Main Bawana Road, Delhi – 110042

CERTIFICATE BY THE SUPERVISOR

Certified that **Aditi** (23/MSCMAT/58) have carried her research work presented in this thesis entitled **"RECENT APPLICATIONS OF PCA AND SVD"** for the award of **Master of Science** from Department of Applied Mathematics, Delhi Technological University, Delhi, under my supervision. The thesis embodies results of original work, and studies are carried out by the students themselves and the contents of the thesis do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University.

Signature

(Mr. Jamkhongam Touthang)

(Department of Applied Mathematics)

Date:

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur , Main Bawana Road, Delhi – 110042

ACKNOWLEDGEMENT

We want to express our appreciation to Mr. Jamkhongam Touthang, Department of Applied Mathematics, Delhi Technological University, New Delhi, for his careful and knowledgeable guidance, constructive criticism, patient hearing, and kind demeanour throughout our ordeal of the present report. We will always be appreciative of his kind, helpful and insightful advice, which served as a catalyst for the effective completion of our dissertation report.

We are grateful to our Department of Mathematics for their continuous motivation and involvement in this project work. We are also thankful to all those who, in any way, have helped us in this journey. Finally, we are thankful to the efforts of our parents and family members for supporting us with this project.

Abstract

In this thesis, Principal Component Analysis is a powerful dimensionality reduction technique that transforms high-dimensional data into a lower-dimensional space while preserving variance. By computing the covariance matrix and its eigenvectors, PCA finds principal components that best represent the data. It is used on large scale in image compression, face recognition, and feature extraction, simplifying complex datasets without losing critical information. SVD is a matrix factorization method that decomposes any matrix into three distinct matrices: $A = U\Sigma V^T$. This decomposition reveals hidden patterns in data and has applications in data compression, noise reduction, and recommendation systems. Unlike PCA, which relies on eigenvectors of the covariance matrix, SVD works directly on the data matrix, making it more versatile. PCA and SVD are two fundamental techniques in linear algebra that have revolutionized data science, machine learning, and image processing. This presentation explores their mathematical foundations, geometric interpretations, and real-world applications.

Keywords:- Dimensionality reduction, Covariance matrix, Directional vector, Matrix factorization, Data compression.

Contents

Chapter 1

Historical Background of PCA and SVD

The story of PCA and SVD is a testament to the evolution of mathematical thought and its practical implications. Emerging from geometric intuition and abstract algebra, both PCA and SVD have grown into powerful tools for data analysis, largely due to the joint contributions of statisticians, mathematicians, and computer scientists over the last 150 years.

1.1 PCA

1.1.1 Origins in Geometry and Statistics

PCA was formally found by the British statistician **Karl Pearson** in his 1901 paper named *"On Lines and Planes of Closest Fit to Systems of Points in Space"*. He sought a way to decrease the dimensionality of multivariate data while preserving its most informative aspects. His approach was geometric—identifying directions (principal axes) along which data varied most.

Three decades later, American statistician **Harold Hotelling** provided a statistical reinterpretation of Pearson's idea. In his 1933 work, Hotelling used the eigenvalue decomposition of the covariance matrix to derive non-correlated variables, now known as principal components. This helped integrate PCA into psychometrics, econometrics, and social science research.

1.1.2 PCA in the Era of Digital Computation

With the rise of computers in the mid-20th century, PCA became increasingly popular for practical applications. Scientists could analyze large datasets with many variables, using PCA to reduce noise and reveal latent structure. Researchers such as **Henry Kaiser** contributed refinements like varimax rotation to enhance interpretability, particularly in factor analysis.

The rise of multivariate statistics in the 1960s and 70s further cemented PCA's status. By then, it had become a standard tool in fields like biology, linguistics, economics, and geology.

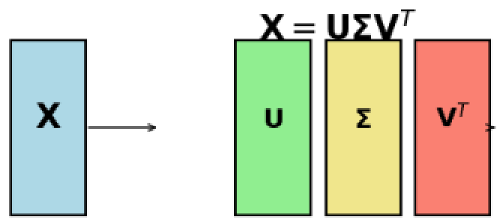

Figure 1.1: Relationship between PCA and SVD: SVD of centered data matrix yields principal components.

1.2 SVD

1.2.1 19th-Century Foundations

SVD's conceptual roots trace back to the works of **Eugenio Beltrami** and **Camille Jordan** in the 1870s. These mathematicians studied quadratic forms and canonical representations, particularly focusing on orthogonal transformations and invariant theory—topics that underpin modern matrix decompositions.

In 1907, the German mathematician **Erhard Schmidt** introduced what we now recognize as the Schmidt decomposition (closely related to SVD) in the context of Hilbert spaces. His ideas anticipated the functional-analytic interpretations of matrix factorizations in infinite dimensions.

1.2.2 20th-Century Computational Breakthroughs

SVD as a practical matrix factorization was only realized with the advent of numerical linear algebra in the mid-20th century. In the 1950s and 60s, **Alston S. Householder** developed methods such as the Householder transformation, which enabled stable numerical computation of matrix decompositions.

The pivotal moment came with the work of **Gene H. Golub** and **William Kahan** in 1965. They devised a numerically stable algorithm for computing the SVD of any rectangular matrix, laying the foundation for its use in scientific computing. This made SVD applicable not only in solving linear systems and matrix approximation but also in applications like least squares, data compression, and signal processing.

1.3 Bridging PCA and SVD

A major insight that shaped modern data analysis was that PCA could be computed using SVD. If a dataset is represented as a matrix $\mathbf{X}$ (with centered columns), then the right singular vectors of $\mathbf{X}$

Table 1.1: Comparative Historical Milestones of PCA and SVD

Year	**PCA Milestone**	**SVD Milestone**
1901	Karl Pearson introduces PCA as geometric projection maximizing variance	—
1933	Hotelling formalizes PCA using covariance matrix and eigenvalue decomposition	—
1958	PCA gains popularity in psychology and factor analysis	Householder develops transformation for orthogonalization
1965	—	Golub and Kahan design stable numerical SVD algorithm
1980s	PCA used in face recognition and multivariate analysis	SVD adopted in signal processing and noise reduction
1990s	Kernel PCA developed for nonlinear transformations	SVD used in Latent Semantic Analysis (LSA) for NLP
2000s	PCA becomes a core ML technique (Hastie et al.)	SVD powers collaborative filtering and recommender systems
Present	PCA applied in genomics, finance, image compression	SVD used in deep learning, matrix completion, tensor decompositions

correspond to the principal directions, and the squares of the singular values correspond to the variances of the principal components.

This connection allowed PCA to scale with numerical stability, and led to its integration into a variety of fields where large-scale data processing was critical. Theoretical advances combined with efficient numerical libraries like LAPACK and LINPACK made PCA–SVD pipelines commonplace in practice.

1.4 Expansion into Machine Learning(ML) and Data Science

In the 21st century, PCA and SVD became central tools in modern data science, especially for dimensionality reduction, noise filtering, and feature extraction. In their seminal book *The Elements of Statistical Learning* (2001), **Hastie**, **Tibshirani**, and **Friedman** presented PCA as one of the most effective unsupervised learning methods. Around the same time, in fields like bioinformatics and natural language processing, SVD-based techniques like latent semantic analysis (LSA) were applied to extract structure from large text and gene expression data.

SVD also became the backbone of recommendation systems. The famous *Netflix Prize* (2006) highlighted SVD's relevance in collaborative filtering, demonstrating its power in reducing sparsity and improving prediction accuracy in user-item rating matrices.

1.5 Legacy and Continuing Impact

Today, PCA and SVD are deeply embedded in curricula across mathematics, statistics, computer science, and engineering. From classical uses in psychometrics to modern applications in deep learning, these tools remain indispensable. They continue to inspire new techniques, such as randomized SVD, kernel PCA, robust PCA, and non-negative matrix factorization.

Their long-standing history, from Pearson and Jordan to Golub and beyond, reflects a rare convergence of pure theory and practical utility—making PCA and SVD enduring pillars of modern data science.

PCA and SVD have evolved from foundational concepts in statistics and linear algebra to essential tools in modern data science. Rooted in early 20th-century mathematical theories by Pearson and Eckart-Young, their development reflects the growing need to simplify and interpret high-dimensional data. Over time, they have proven remarkably versatile, influencing fields such as signal processing, machine learning, and computational biology. Understanding their historical development highlights the enduring power of linear algebra in extracting structure, reducing complexity, and enabling innovation across scientific and technological domains.

Chapter 2

Mathematical Foundations

2.1 Overview

PCA and SVD are rooted in core concepts of linear algebra. These include vector spaces, inner products, orthogonality, and spectral theory. This chapter provides a theoretical framework for understanding these techniques by examining the essential mathematical structures that support them.

2.2 Vector Spaces and Subspaces

1. A **vector space** V over a field $\mathbb{F}$can be defined as a set of elements, called vectors, which follows two operations: that are vector addition and scalar multiplication:

   - Additive closure and scalar closure,
   - Satisfies associativity and commutativity while adding vectors,
   - There exists a zero vector and an additive inverse,
   - Distributivity of S.M over both scalar and V.A.

   The set $\mathbb{R}^n$ under element-wise V.A and S.M is a standard example of a real vector space.

2. A Subspace W can be defined as if W is itself a V.S under the inherited operations from V. Generely, this requires that W contains the zero vector and is closed under both V.A and S.M.

   A collection of vectors $\{v_1, \ldots, v_k\}$ in $\mathbb{R}^n$ generates a subspace, denoted $\text{span}\{v_1, \ldots, v_k\}$.

2.3 Representation of Matrix and Linear maps

Let $U(F)$ and $V(F)$ are two V.S over the same field $\mathbb{F}$. A mapping $f : U \to V$ is said to be a **linear mapping** or transformation of U into V if for all $u, v \in V$ and all scalars $\alpha \in \mathbb{F}$:

f(u + v) = f(u) + f(v), $f(\alpha u) = \alpha f(u)$.

Every linear mapping or transformation f :R^n →R^m can be represented by an $m \times n$ matrix A, such that $f(x) = Ax$.

2.4 Inner Product Spaces and Orthogonality

An **inner product** on a vector space V(R) a function $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{R}$ that satisfies:

- Symmetry: $\langle x, y\rangle = \langle y, x\rangle$,
- Linearity:for all α, β, γ belongs to V and a,b belongs to F implies :
  $\langle a\alpha$ + bβ, γ $\rangle$ = a $\langle \alpha$,γ $\rangle + b\langle \beta$,γ $\rangle$
  $NonNegativity$:$\langle x, x\rangle \geq 0$ if and only if $x = 0$.

In $\mathbb{R}^n$, the inner product is given by $\langle x, y\rangle$ = (x$_1 y_1$) + ($x_2 y_2$) +($x_3 y_3$) +...+ ($x_n y_n$)
x and y will be **orthogonal** if their inner product is 0.

2.5 Eigenstructure of Matrices

If there is a square matrix $A \in \mathbb{R}^{n\times n}$, a scalar k ,a non-zero vector v which satisfies the:

$$Av = kv.$$

Then k be an eigenvalue of A, and v is the associated eigen structure.

Theorem 1 (Spectral Theorem)**.** *Let $A = A^T$ and $A \in R.Then$:*
1)All A has real eigenvalues,
2) and also has orthonormal basis of eigenvectors for $\mathbb{R}^n$,

2.6 SVD

Theorem 2 (SVD)**.** *Let A be a matrix and $A \in \mathbb{R}^{m\times n}$. Then A admits a decomposition:*

$$A = U\Sigma V^T,$$

where:

- *U and V are orthogonal matrices which belongs to $\mathbb{R}^{m\times m}$, $\mathbb{R}^{n\times n}$ respectively.*
- *Σ is diagonal belongs to $\mathbb{R}^{m\times n}\sigma_1 \geq \cdots \geq \sigma_r > 0$ on the diagonal (called singular values).*

Singular values we obtained are square roots of the non-zero eigenvalues of $A^T A$ and AA^T:

$$\sigma_i = \sqrt{\lambda_i(A^T A)}.$$

2.7 PCA

2.7.1 PCA as Variance Maximization

PCA identifies orthogonal directions by which the data variance is maximum. We have a centered data matrix $X \in \mathbb{R}^{n \times p}$, the method proceeds as follows:

1. Compute the empirical covariance matrix: $C = \frac{1}{n-1} X^T X$.
2. Solve for eigenvalues and eigenvectors: $C v_i = \lambda_i v_i$.
3. Retain the first k eigenvectors corresponding to the highest eigenvalues.
4. Projection of the data: $Z = X V_k$, with $V_k = [v_1, \ldots, v_k]$.

2.7.2 SVD Perspective on PCA

SVD of centered matrix X is:

$$X = U \Sigma V^T.$$

Here, columns of V serve as the principal directions. The relationship between singular values σ_i and eigenvalues λ_i of the covariance matrix is:

$$\lambda_i = \frac{\sigma_i^2}{n-1}.$$

Chapter 3

Algorithms

3.1 PCA and SVD for Compression of Image

Compressing image is an important application of dimensionality reduction, and two widely utilized techniques in this domain are **PCA** and the **SVD**. This portion outlines and compares the algorithms of PCA and SVD when applied to grayscale image compression.

3.1.1 Image Compression by PCA

PCA is a method that reorients the dataset in a new coordinate system, prioritizing directions through which the data varies the most.

Consider $A \in \mathbb{R}^{m \times n}$ represent a grayscale image matrix, where each row corresponds to a row of pixel intensities, and each column denotes pixel intensity across m observations. The PCA-based compression can be described through the following steps:

Step 1: Normalize Pixel Values

Scale all pixel intensities to the range $[0, 1]$ by dividing the matrix by 255:

$$A \leftarrow \frac{A}{255}$$

Step 2: Center the Data

Compute the mean across rows for each column:

$$\mu = \frac{1}{m} \sum_{i=1}^{m} A_i$$

Subtract the mean vector from every row:

$$A_{\text{centered}} = A - \mu$$

Step 3: Construct Covariance Matrix

Covariance matrix, that shows the joint variability of pixels, is given by:

$$C = \frac{1}{m-1} A_{\text{centered}}^T A_{\text{centered}} \in \mathbb{R}^{n \times n}$$

Step 4: Eigen Decomposition

Compute eigenvalues and eigenvectors of the covariance matrix:

$$CV = V\Lambda$$

where V contains the eigenvectors and Λ is a diagonal matrix of eigenvalues.

Step 5: Choose k Principal Components

Select the top k eigenvectors corresponding to the k highest eigenvalues:

$$V_k \in \mathbb{R}^{n \times k}$$

Step 6: Projection and Reconstruction

Project the centered data onto the chosen components:

$$Z = A_{\text{centered}} V_k$$

Reconstruct the approximate image from the reduced representation:

$$A_{\text{PCA}} = Z V_k^T + \mu$$

Clip and rescale A_{PCA} to return pixel values to the range $[0, 255]$ if needed.

Goal: PCA seeks to retain the most significant variance using fewer dimensions:

$$\max \operatorname{Var}(Z) = \operatorname{tr}(V^T C V), \quad \text{with } V^T V = I$$

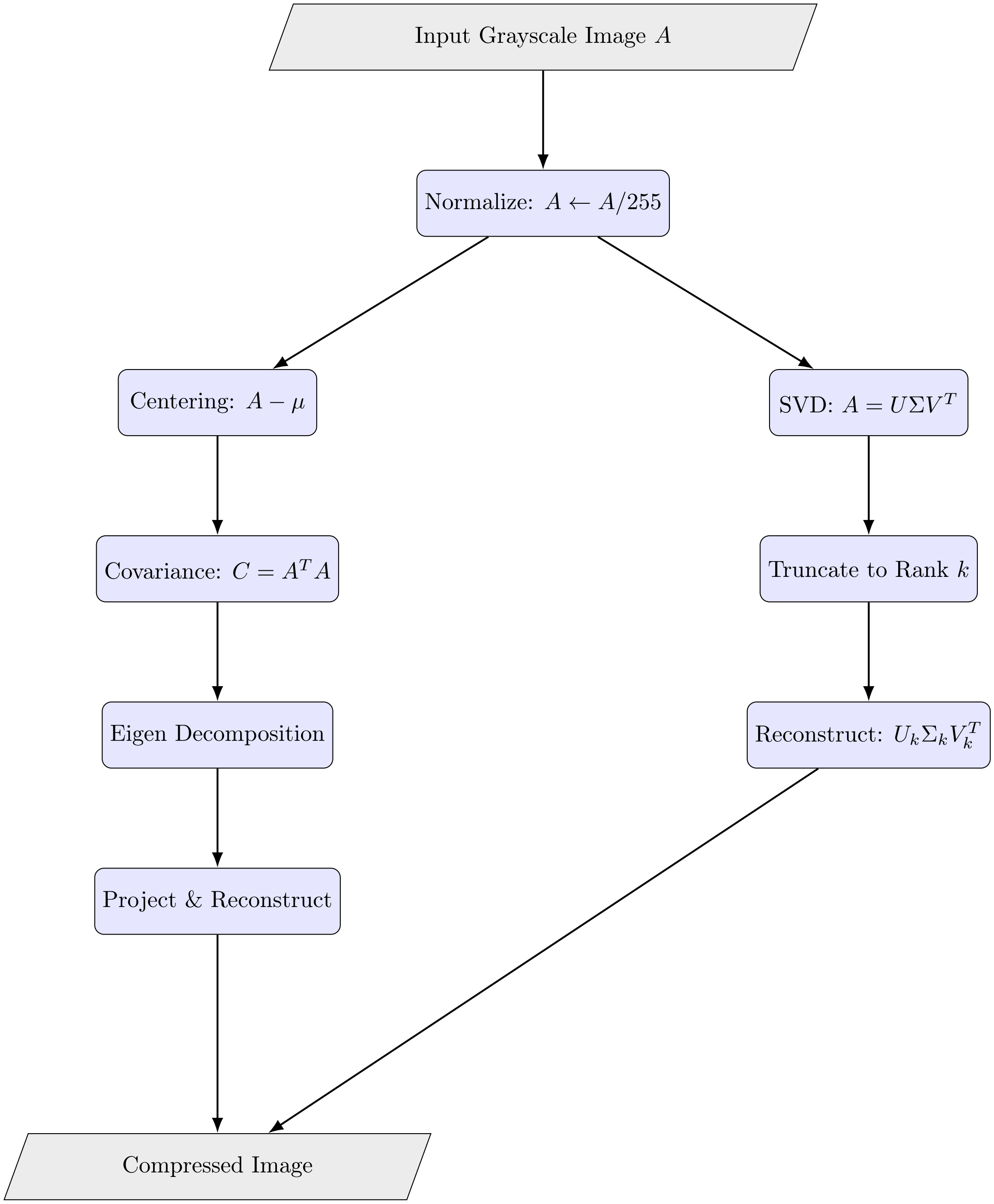

Figure 3.1: Workflow of PCA and SVD

3.1.2 Compressing image by SVD

SVD decomposes a matrix A into three matrices:

$$A = U\Sigma V^T$$

Here, U and V are orthogonal matrices, which belongs to $\mathbb{R}^{m\times m}$, $\mathbb{R}^{n\times n}$ respectively and $\Sigma \in \mathbb{R}^{m\times n}$ is a diagonal matrix of singular values arranged in decreasing order.

Step 1: Normalize

Convert pixel intensities to a floating-point range:

$$A \leftarrow \frac{A}{255}$$

Step 2: Find the SVD

Factor the matrix of image:

$$A = U\Sigma V^T$$

Step 3: Truncate the Decomposition

Retain only the first k singular values and the corresponding singular vectors:

$$A_k = U_k \Sigma_k V_k^T$$

with:

- $U_k \in \mathbb{R}^{m\times k}$
- $\Sigma_k \in \mathbb{R}^{k\times k}$
- $V_k \in \mathbb{R}^{n\times k}$

Step 4: Reconstruct the Image

Rebuild the compressed image:

$$A_{\text{SVD}} = U_k \Sigma_k V_k^T$$

Ensure values lie within the valid image range before displaying or storing.

Goal: SVD attains the best rank-k approximation in terms of Frobenius norm:

$$\min \|A - A_k\|_F^2 = \sum_{i=k+1}^{\min(m,n)} \sigma_i^2$$

3.1.3 Comparative Approaches and Optimization

Using SVD for PCA

PCA can be performed efficiently by computing the SVD of the centered matrix:

$$A_{\text{centered}} = U\Sigma V^T$$

Principal components are the columns of V, and the eigenvalues of the covariance matrix are:

$$\lambda_i = \frac{\sigma_i^2}{m-1}$$

Randomized SVD

For large datasets, traditional SVD is computationally expensive. Randomized SVD accelerates the process by projection of data onto a lower-dimensional subspace using random matrices:

1. Draw a random Gaussian matrix $\Omega \in \mathbb{R}^{n\times(k+p)}$
2. Find the sample matrix $Y = A\Omega$
3. Use QR decomposition to obtain an orthonormal basis Q from Y
4. Form a smaller matrix $B = Q^T A$
5. Find the SVD: $B = \hat{U}\Sigma V^T$
6. Approximate left singular vectors: $U_k = Q\hat{U}$

This method lowers the computational cost to approximately $O(mn \log k)$.

Incremental SVD

When data arrives sequentially (e.g., in streaming settings), incremental SVD allows updating the decomposition without reprocessing the entire dataset:

$$A_{t+1} = \begin{bmatrix} A_t \\ x_{t+1}^T \end{bmatrix} \approx U_{t+1}\Sigma_{t+1}V_{t+1}^T$$

This technique is mainly valuable in conditions where memory or processing time is limited.

In summary, PCA and SVD serve as foundational tools in image compression. While PCA is ideal for analyzing variance and projecting onto informative axes, SVD offers optimal low-rank reconstructions. For large-scale problems, optimized versions such as randomized or incremental methods enable scalable implementations.

Aspect	PCA	SVD
Core Idea	Projects data onto orthogonal components that capture maximum variance	Factorizes matrix into singular vectors and singular values
Computation	Requires covariance matrix and eigen decomposition	Direct matrix factorization
Optimality	Maximizes retained variance	Minimizes Frobenius norm error
Efficiency	Faster for smaller datasets	More efficient with optimized libraries for large data
Scalability	Requires full matrix; not ideal for streaming	Supports incremental and randomized variants
Use Case	Good for exploratory data analysis and interpretation	Preferred for optimal low-rank approximations

Table 3.1: Comparison of PCA and SVD for Image Compression

Chapter 4

Applications for PCA and SVD

The rapid increase of applications of PCA and SVD in diverse Engineering fields has been driven by the surge in data-intensive problems. From image and signal processing in electrical engineering to structural health monitoring in civil engineering and fault detection in mechanical systems, these techniques offer robust solutions for optimizing performance, reducing computational load, and improving model accuracy.

4.1 Applications of PCA

1. **Material Sciences**:Material science studies the properties and behaviors of materials like metals, ceramics, and polymers. In this field, **PCA** , a linear algebra technique, helps analyze complex datasets from experiments (e.g., stress-strain data or microstructure images). PCA reduces dimensionality by converting correlated variables into uncorrelated principal components, highlighting key patterns and variations in material properties. This simplifies understanding of factors affecting strength, durability, or conductivity. By applying eigenvalues and eigenvectors from covariance matrices, PCA reveals dominant features in materials, aiding in design, quality control, and predicting performance efficiently.

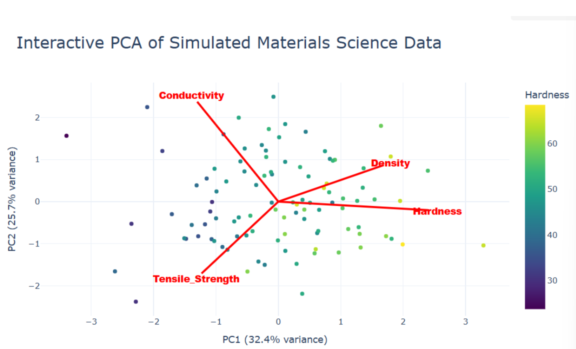

Figure 4.1: Interative PCA if Simulated Materials Science Data

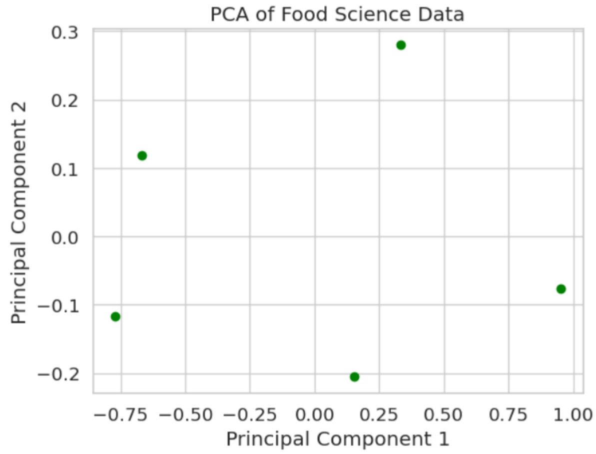

Figure 4.2: Measured Features of Food Samples Used for PCA Analysis

2. **Food Science and Technology**: **PCA** uses linear algebra to analyze complex data from food composition, quality, and sensory evaluations. By transforming correlated variables into principal components using eigenvalues and eigenvectors of the data's covariance matrix, PCA reduces dimensionality and reveals key factors affecting taste, texture, or shelf life. This helps identify patterns, classify food types, and monitor quality control efficiently. PCA aids in optimizing formulations, detecting adulteration, and improving processing methods by focusing on the most important variations in food data, making it a powerful tool for research and industry applications.

Table 4.1: Hypothetical Food Science Data: Samples and Measured Features

Sample	Sugar Content	Acidity	Texture Score	Moisture (%)
1	10.5	7.8	5.2	12.3
2	11.2	8.1	5.0	11.8
3	9.8	7.5	5.4	12.5
4	10.1	7.9	5.3	12.0
5	11.0	8.2	5.1	11.7

3. **Fault Diagnosis**:Fault diagnosis using PCA applies linear algebra to detect anomalies in complex systems by analyzing multivariate data. PCA reduces dimensionality by converting correlated variables into uncorrelated principal components using eigenvalues and eigenvectors of the covariance matrix. In fault diagnosis, normal system behavior is captured by the principal components with the greatest variance. Any deviations, or faults, appear as residuals in the lower-variance components. By monitoring these residuals, PCA helps identify abnormal patterns or sensor failures. This approach is widely used in industries for condition monitoring, enhancing reliability, and enabling early detection of system faults through mathematical rigor.

Table 4.2: Description of Simulated Sensor Data for Fault Diagnosis

Parameter	Description
Number of Samples	120 (100 normal, 20 faulty)
Number of Sensors	4
Fault Location	Sensor 2 (Index 1)
Fault Type	Mean shift (+5 units)
Detection Metric	Squared Prediction Error (SPE)
PCA Components Used	Top 2 principal components
Threshold	95^{th} percentile of normal SPE

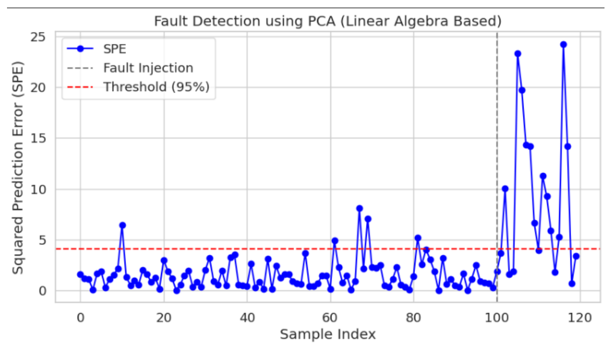

Figure 4.3: Caption

4. **Drug Discovery and Biomedical Data**:In drug discovery and biomedical data analysis, **PCA (Principal Component Analysis)** applies linear algebra to simplify high-dimensional datasets such as gene expression, protein interactions, or compound activity. Using eigenvalues and eigenvectors of the data's covariance matrix, PCA transforms correlated biological features into uncorrelated principal components. This reduces noise and highlights patterns that reveal potential drug targets or biological responses. By identifying dominant variations, PCA helps in clustering diseases, classifying compounds, and visualizing complex biomedical relationships. Its foundation in linear algebra enables efficient data compression and interpretation, making it a powerful tool in bioinformatics and pharmaceutical research.

Table 4.3: Simulated Biomedical Dataset Used for PCA

Parameter	Description
Samples	100 simulated patient or compound profiles
Features	20 (e.g., gene expression levels or molecular descriptors)
Informative Features	10 strongly related to class distinction
Redundant Features	5 (linear combinations of informative ones)
Target Classes	3 (e.g., drug response groups or disease types)
Analysis Method	PCA using eigen decomposition
Purpose	Visualize patterns and reduce dimensionality

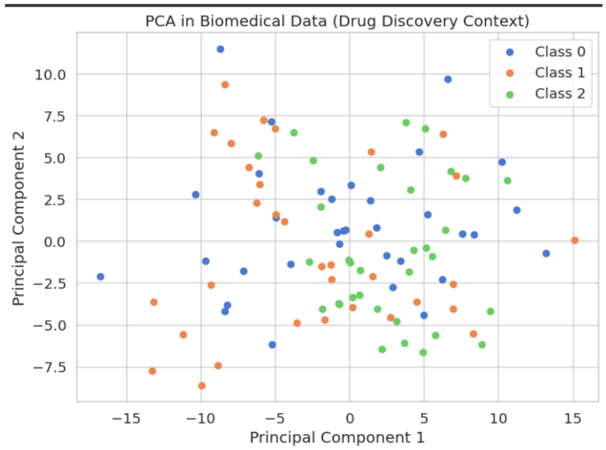

Figure 4.4: Fault Detection using PCA (Linear Algebra Based

5. **Quantify Technique in Sports**:In sports, **PCA** is used to quantify athletic performance and technique by analyzing high-dimensional data such as motion capture, biomechanics, or training metrics. Using linear algebra, PCA converts correlated variables into uncorrelated principal components through eigenvalue decomposition of the covariance matrix. This highlights dominant movement patterns and removes noise or redundancy. Coaches and analysts use PCA to compare athletes, detect inefficiencies, and improve technique by focusing on the most significant variations. It simplifies complex datasets into interpretable components, making performance assessment and injury prevention more efficient through the mathematical principles of dimensionality reduction and vector space analysis.

Table 4.4: Description of Simulated Sports Performance Dataset for PCA Analysis

Parameter	**Description**
Number of Samples	60 motion samples (20 samples per athlete)
Number of Athletes	3 (simulating varying levels of technique)
Motion Features	6 features (e.g., joint angles, speed, acceleration, force)
Athlete 1	High-performing athlete (data centered at +0.5)
Athlete 2	Moderate-performing athlete (data centered at 0.0)
Athlete 3	Lower-performing athlete (data centered at –0.5)
Noise Level	Gaussian noise with standard deviation of 0.1 to simulate variability
Purpose	Evaluate and visualize differences in sports technique using PCA

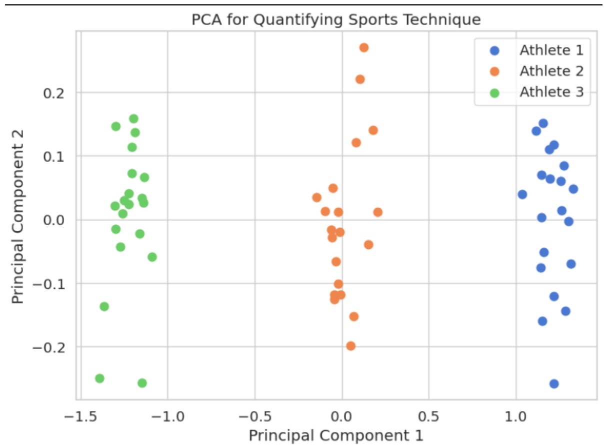

Figure 4.5: PCA for Quantifying Sports Technique

4.2 APPLICATIONS OF SVD

1. **Solving Linear Least-Square Problems**:SVD is crucial for solving linear least-squares problems, especially when systems are overdetermined or matrices are ill-conditioned. By decomposing a matrix into singular values and orthogonal vectors, SVD provides a stable and robust method to find approximate solutions when direct inversion fails or is inaccurate. This decomposition helps to isolate the effect of small singular values that can cause numerical instability. SVD's ability to handle rank-deficient matrices makes it indispensable in data fitting, image reconstruction, and scientific computations where precise solutions are necessary despite noisy or incomplete data.

   Given an overdetermined system $A\mathbf{x} = \mathbf{b}$, where

$$A = \begin{bmatrix} 2 & 1 \\ 1 & 3 \\ 0 & 4 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 7 \\ 10 \\ 8 \end{bmatrix},$$

   the objective is to find $\mathbf{x}$ that minimizes the residual $\|\mathbf{b} - A\mathbf{x}\|$.

   Using SVD, A is decomposed as

$$A = U\Sigma V^T,$$

   where U and V are orthogonal matrices, and Σ is a diagonal matrix with singular values. The least squares solution is then computed as

$$\mathbf{x} = V\Sigma^{-1}U^T\mathbf{b}.$$

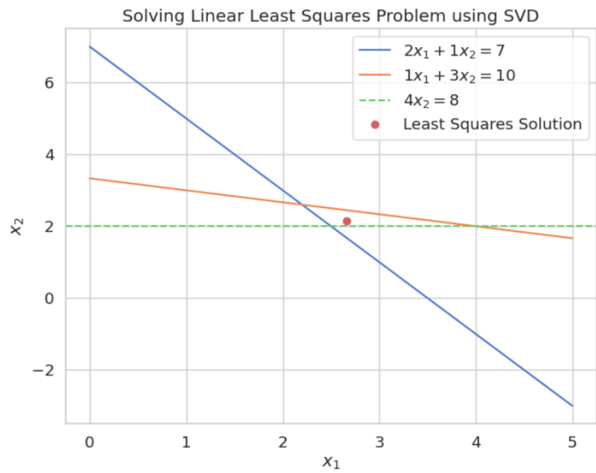

Figure 4.6: Solving Linear Least-Square Problem using SVD

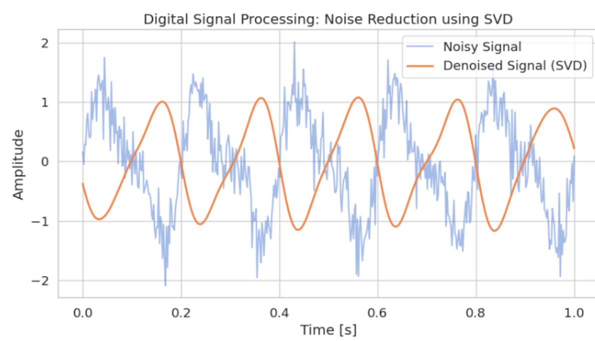

Figure 4.7: Digital Signal Processing: Noise Reduction using SVD

2. **Digital Signal Processing**: SVD can be used to reduce noise and enhance signal quality. A noisy signal is converted into a Hankel matrix, which organizes overlapping segments of the signal. The Hankel matrix H is decomposed via SVD into orthogonal matrices U, V^T, and a diagonal matrix of singular values Σ:

$$H = U\Sigma V^T$$

The singular values represent the energy contained in each orthogonal component of the signal. By retaining only the top k singular values and setting the rest to zero, we filter out noise components associated with smaller singular values. Reconstructing the Hankel matrix from the truncated decomposition yields a denoised signal upon averaging the anti-diagonals of the matrix.

3. **Generalize Eigenvalue Problems**:SVD generalizes eigenvalue problems by decomposing any rectangular matrix into singular values and vectors, extending beyond square matrices where eigenvalues are defined. This decomposition provides a more numerically stable and versatile framework for analyzing data structure. It is extensively used in Principal Component Analysis (PCA), dimensionality reduction, and system stability analysis by identifying dominant modes of

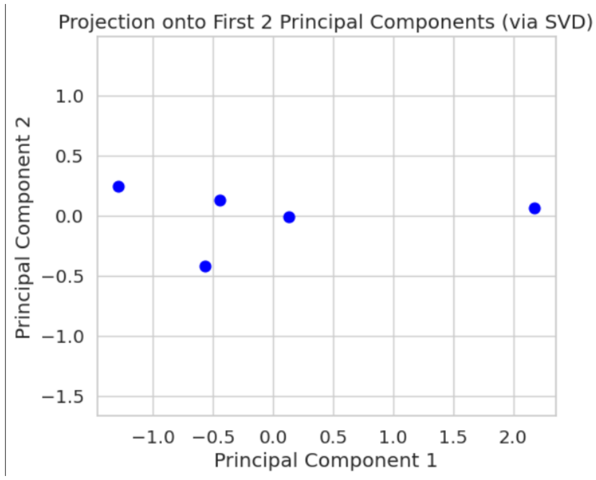

Figure 4.8: Projection onto First 2 Principal Components(via SVD)

variation or system behavior. Unlike traditional eigenvalue methods, SVD handles rank-deficient or non-square matrices gracefully, making it essential in applied mathematics, control theory, and machine learning for understanding and simplifying complex datasets.

Table 4.5: Sample Data Matrix (5 Observations, 3 Features)

Sample	Feature 1	Feature 2	Feature 3
1	2.5	2.4	1.0
2	0.5	0.7	1.1
3	2.2	2.9	0.9
4	1.9	2.2	1.2
5	3.1	3.0	1.3

4. **Data Science**:In this, SVD is a cornerstone method for dimensionality reduction, feature extraction, and noise filtering in large datasets. It converts high-dimensional data into a lower-dimensional space while preserving key information, which improves the performance and speed of machine learning algorithms. SVD is fundamental in collaborative filtering for recommendation systems, such as those used by streaming platforms, by identifying latent user-item relationships. It is also used in Latent Semantic Analysis (LSA) for natural language processing to uncover hidden semantic structures in text data, facilitating efficient search, classification, and topic modeling.

Table 4.6: Document-Term Matrix for SVD-based Feature Extraction

Document	book	novel	apple	fruit	banana
Doc1	1	1	0	0	0
Doc2	3	3	0	0	0
Doc3	0	0	4	4	1
Doc4	0	0	5	5	2
Doc5	0	1	0	0	0

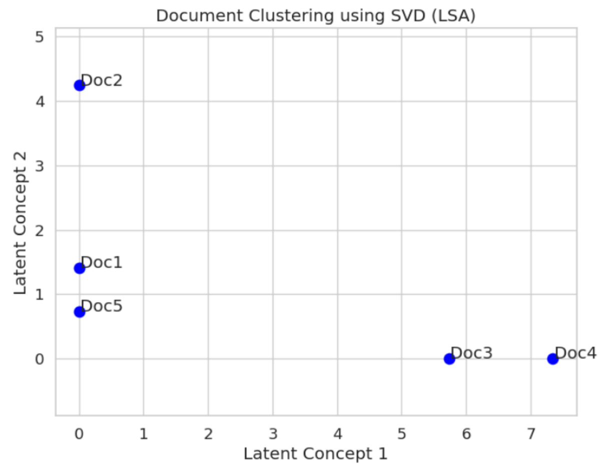

Figure 4.9: Document Clustering using SVD (LSA)

5. **Fault Diagnosis**:SVD plays a vital role in fault diagnosis by analyzing sensor data from complex industrial systems. By decomposing data matrices into singular values, SVD helps isolate patterns that signify abnormalities or faults, even in the presence of noise and measurement errors. This capability allows early detection of failures, reducing downtime and maintenance costs. It supports predictive maintenance by highlighting key fault-related features, enabling timely interventions in manufacturing, aerospace, and energy sectors. SVD enhances system reliability and safety by providing a mathematically rigorous method to monitor and diagnose faults from high-dimensional sensor data effectively.

Table 4.7: Sample Sensor Data for Fault Diagnosis

Time	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Sensor 5
1	0.49	-0.14	0.65	1.52	-0.23
2	-0.23	0.72	-0.21	0.87	0.38
...	...	...	...	...	...
51	-0.43	0.87	**6.89**	-0.23	0.48
52	0.26	-0.12	**7.12**	0.21	-0.87

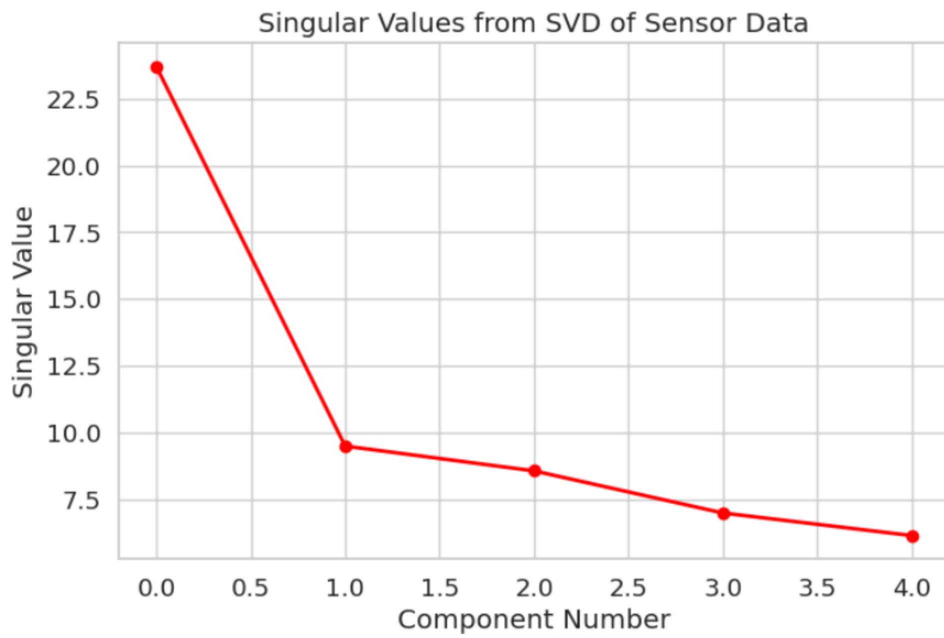

Figure 4.10: Singular Values from SVD of Sensor Data

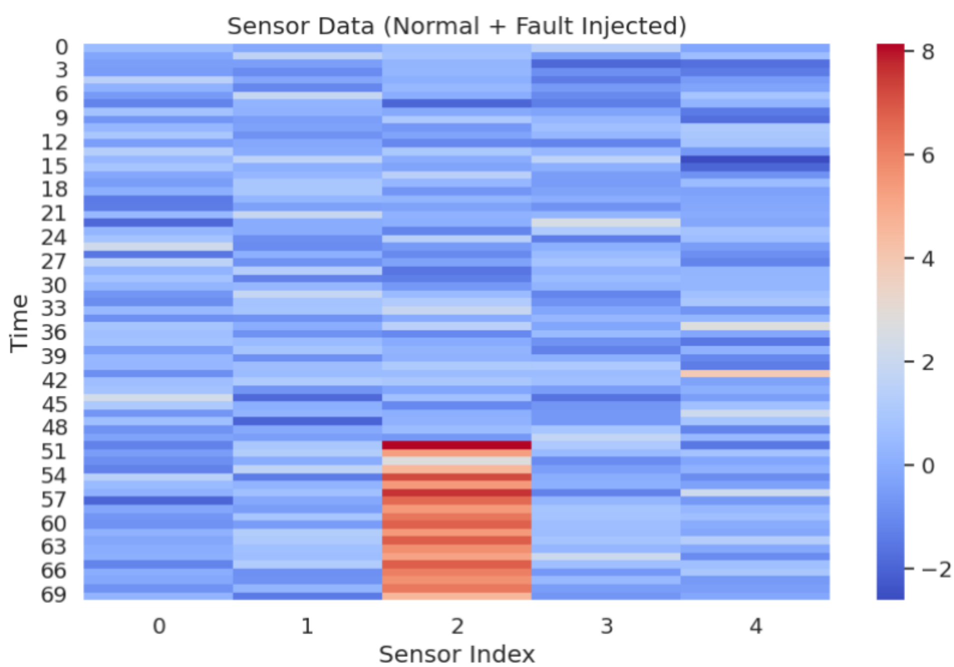

Figure 4.11: Sensor Data (Normal + Fault Injected)

Chapter 5

Future Directions

PCA and SVD are key linear algebra tools for simplifying data by finding linear patterns. However, many real-world datasets have complex nonlinear structures that these methods cannot capture well. Future directions focus on nonlinear extensions like Kernel PCA and autoencoders, which use kernels or neural networks to uncover hidden patterns beyond linear assumptions. Additionally, improving scalability with randomized algorithms and distributed computing helps handle large datasets efficiently. Combining linear algebra with modern machine learning and optimization techniques promises more powerful, flexible dimensionality reduction tools suited for complex, high-dimensional data in fields like bioinformatics, image processing, and AI.

5.1 Non-Linear Methods for PCA and SVD: Future Directions in Linear Algebra

Traditional PCA and SVD are fundamentally linear techniques. They project data onto linear subspaces using eigenvectors or singular vectors of the covariance or data matrix. However, real-world data often resides on non-linear manifolds embedded in high-dimensional spaces. To capture these complexities, non-linear extensions of PCA and SVD have been developed, paving the way for future advances in linear algebra and its applications.

5.1.1 Kernel PCA

Kernel PCA extends linear PCA by using a kernel function for mapping data into a high-dimensional feature space, where linear PCA is then applied. This thing captures non-linear relationships in the original data. It is particularly useful in fields such as image recognition and bioinformatics.

Future Direction: The development of adaptive and learning-theoretic kernels to better model specific data geometries.

Table 5.1: Sample of Swiss Roll Dataset (first 10 points)

X	**Y**	**Z**
6.752	11.161	1.166
5.858	12.237	4.889
5.574	9.737	4.182
7.459	7.487	2.657
6.865	10.214	5.671
6.045	11.852	0.944
5.433	11.143	1.555
6.927	10.478	0.616
6.127	9.345	3.242
6.243	12.938	1.032

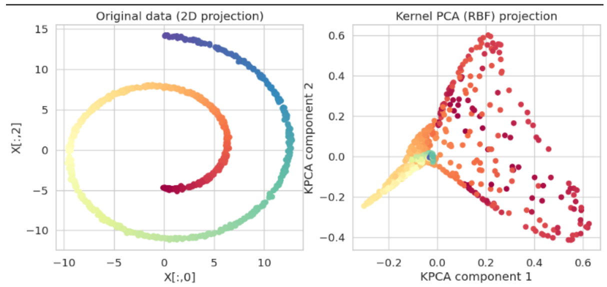

Figure 5.1: Kernal PCA

5.1.2 Autoencoders and Deep PCA

Autoencoders are neural networks that learn non-linear transformations by minimizing reconstruction error. Deep PCA variants extract latent features from high-dimensional data and are robust to noise and outliers. Variational autoencoders further add a probabilistic interpretation to the embedding.

Future Direction: Integration with geometric deep learning and differentiable algebra for learning non-linear latent structures.

Table 5.2: Sample of S-Curve Dataset (first 10 points)

X	**Y**	**Z**
-1.452	0.269	1.707
-0.346	0.138	0.364
-1.001	0.285	1.325
-0.054	-0.146	0.440
0.327	0.123	-0.059
-0.138	0.254	0.249
0.144	0.443	-0.142
-0.712	0.225	1.101
1.365	0.082	-1.313
-0.219	0.228	0.577

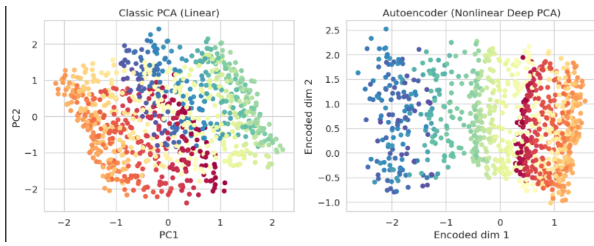

Figure 5.2: Autoencoders and Deep PCA

5.2 Scalability Innovations

As data size and dimensionality grow, classical linear algebra methods like PCA and SVD face heavy computational costs. Future scalability innovations include **randomized algorithms** that approximate decompositions faster by using random projections, reducing complexity significantly. **Incremental and online methods** update factorizations on streaming data without full recomputation. **Distributed and parallel computing** enables processing massive datasets by splitting computations across multiple machines. Additionally, exploiting **low-rank and sparse matrix structures** helps reduce memory and compute demands. These advances make linear algebra techniques efficient and practical for big data, maintaining their importance in modern data science and machine learning.

5.2.1 Simulated Distributed PCA

Simulated distributed PCA splits large datasets across multiple nodes, computes local PCA on each, and then aggregates results. This mimics parallel processing and reduces memory and computation load. It's a scalable approach that enables linear algebra techniques like PCA to handle massive data in distributed systems or real-time analytics.

Table 5.3: Small Sample of Synthetic Gaussian Data

X1	**X2**	**X3**	**X4**	**X5**
0.497	-0.138	0.648	1.523	-0.234
0.542	-0.461	-0.465	0.242	-1.913
-1.412	1.465	-0.225	0.067	0.112
-0.291	-0.234	1.460	-0.845	-0.589
0.010	-0.539	-0.228	-0.483	0.546

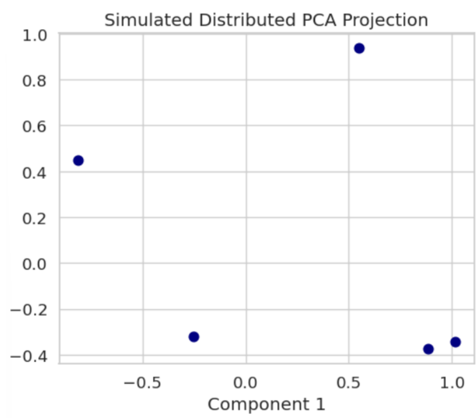

Figure 5.3: Simulated Distributed PCA

5.2.2 Randomized Algorithms

Randomized algorithms, such as randomized SVD or PCA, use random projections to approximate matrix decompositions efficiently. These methods significantly reduce computational complexity while maintaining acceptable accuracy. Ideal for large-scale or streaming data, they represent a future-forward solution to scaling linear algebra operations in data science and machine learning.

Table 5.4: Sample of Swiss Roll Dataset (first 10 points)

X	**Y**	**Z**
-6.048	6.125	-4.509
-6.055	6.542	-4.539
-5.586	6.400	-4.889
-6.511	6.123	-4.028
-5.683	7.519	-4.896
-6.131	5.678	-4.479
-6.087	6.978	-4.625
-5.893	5.720	-4.792
-6.184	5.847	-4.381
-6.124	5.399	-4.480

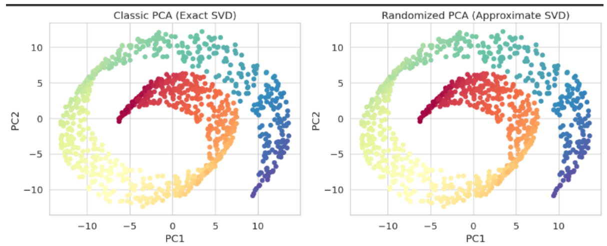

Figure 5.4: Visualization of Randomized Algorithms

5.3 Deep learning

Deep learning advances the future of linear algebra by leveraging large-scale matrix operations, eigen decompositions, and tensor algebra to train neural networks efficiently. Techniques like autoencoders and deep PCA offer nonlinear dimensionality reduction, surpassing traditional linear methods. As models grow, innovations in GPU-accelerated linear algebra and distributed computation will be critical to scale and optimize deep learning architectures across real-world, high-dimensional datasets.

5.3.1 SVD Layers in Neural Networks

An SVD layer in neural networks decomposes weight matrices into singular vectors and values, enabling compression, regularization, and improved interpretability. It reduces model complexity by approximating dense layers with low-rank structures, cutting storage and computation costs. In future directions, SVD layers will support scalable and energy-efficient deep learning by integrating linear algebraic structure directly into neural architectures.

Table 5.5: Input Data for SVD Layer Compression (5 samples $\times$ 5 features)

F1	**F2**	**F3**	**F4**	**F5**
0.139	0.732	0.978	0.365	0.588
0.747	0.615	0.241	0.890	0.349
0.842	0.639	0.390	0.415	0.926
0.313	0.738	0.242	0.495	0.725
0.257	0.586	0.471	0.322	0.100

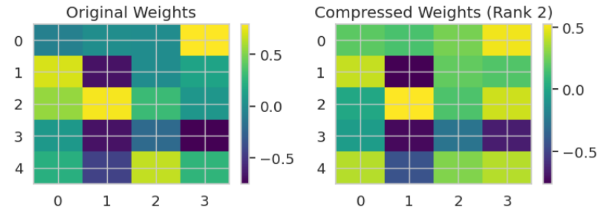

Figure 5.5: Compressed Weight by using SVD Layers

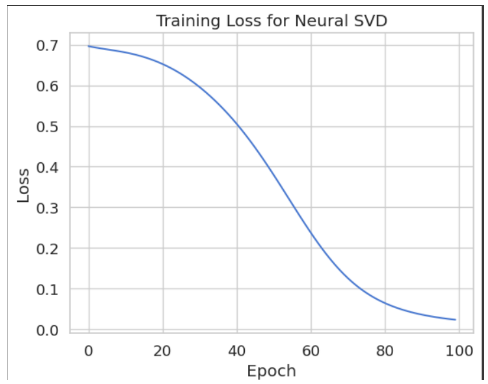

Figure 5.6: Training Loss for Neural SVD

5.3.2 Neural SVD

Neural SVD refers to training deep models that learn optimal low-rank matrix factorizations, extending SVD with nonlinear transformations. It captures latent structure in high-dimensional data more effectively than traditional SVD. As deep learning evolves, neural SVD offers a future-ready framework for embedding linear algebra into neural design, enabling compact models, real-time inference, and deeper insight into learned representations.

Table 5.6: Input Data for Neural SVD Layer (5 samples × 5 features)

F1	**F2**	**F3**	**F4**	**F5**
0.421	-0.273	-1.542	0.143	-0.957
1.392	0.650	-0.478	-0.121	0.205
-0.604	-0.788	0.372	1.501	-0.019
0.899	-0.307	0.010	0.822	-0.541
-0.272	-0.687	-0.097	0.309	0.151

5.4 Emerging Applications

Emerging applications of linear algebra in future directions include quantum computing, deep learning optimization, and large-scale data analytics. Techniques like tensor decompositions and randomized matrix algorithms enable efficient handling of massive datasets. Innovations in linear

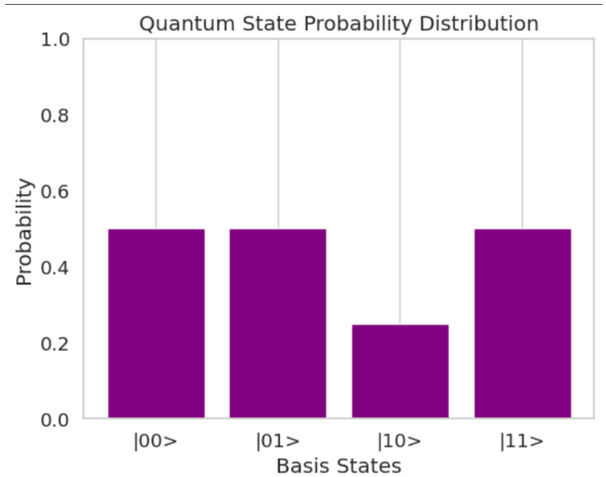

Figure 5.7: Quantum State Probability Distribution

algebra accelerate machine learning model training and enhance precision in simulations. These advancements foster breakthroughs in AI, signal processing, and scientific computing, driving scalable, interpretable, and robust solutions across diverse fields.

Emerging applications include:

5.4.1 Quantum Machine Learning

PCA/SVD principles inform quantum algorithms for dimensionality reduction:

$$|A\rangle = \sum_{i,j} A_{ij}|i\rangle|j\rangle \to \sum_{k=1}^{r} \sigma_k|u_k\rangle|v_k\rangle \tag{5.1}$$

achieving exponential speedup for certain problem classes.

Table 5.7: Quantum State Vector Amplitudes (2-Qubit System)

Basis State	Amplitude (Complex)
$\|00\rangle$	$0.5 + 0.5i$
$\|01\rangle$	$0.5 - 0.5i$
$\|10\rangle$	$0.5 + 0.0i$
$\|11\rangle$	$0.5 - 0.5i$

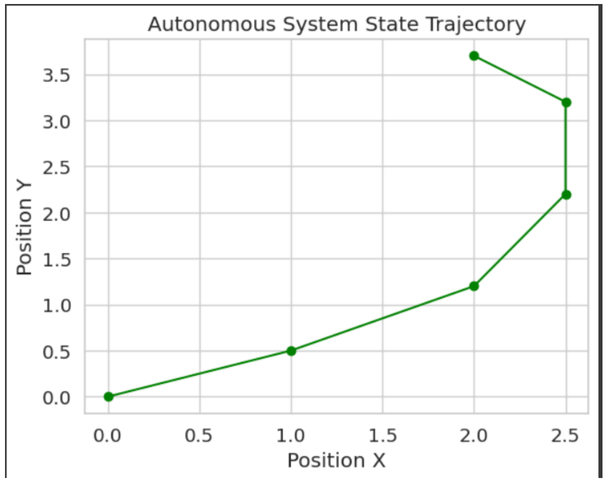

Figure 5.8: Autonomous System State Trajectory

5.4.2 Autonomous Systems

Real-time PCA/SVD powers feature extraction in autonomous vehicles and robots:

$$\Phi_t = \text{Update}(\Phi_{t-1}, x_t) \tag{5.2}$$

where Φ_t represents the current feature space and x_t is new sensory input. Adaptive dimensionality reduction enables efficient operation under varying environmental conditions.

Table 5.8: Control Inputs (Velocity) over 5 Time Steps

Time Step	**Velocity X**	**Velocity Y**
1	1.0	0.5
2	1.0	0.7
3	0.5	1.0
4	0.0	1.0
5	-0.5	0.5

Chapter 6

Conclusion

This dissertation presents a comprehensive study of PCA and SVD, focusing on their mathematical foundations, algorithmic implementations, and wide-ranging applications. Beginning with their historical roots—PCA from Karl Pearson's early 20th-century work and SVD from 19th-century algebra—the thesis illustrates how these tools evolved into vital techniques in modern data analysis.

Mathematically, both PCA and SVD are grounded in linear algebra concepts such as vector spaces, eigenvalues, and orthogonality. PCA converts correlated variables into uncorrelated principal components, effectively reducing dimensionality while preserving variance. SVD, on the other hand, factors matrices into orthogonal components, revealing latent structures and enabling robust approximations of data.

Algorithmic comparisons, particularly in the context of image compression, highlight how PCA maximizes data variance through eigen decomposition, while SVD offers optimal low-rank matrix approximations. Advanced methods like randomized and incremental SVD enhance scalability for large datasets.

The dissertation further explores diverse real-world applications. PCA finds use in material science, food technology, fault diagnosis, biomedical data, and sports analysis. SVD powers solutions in digital signal processing, least-squares problems, recommendation systems, and natural language processing. Together, they underpin many key methods in machine learning and artificial intelligence.

Looking forward, the thesis discusses future directions such as Kernel PCA, deep autoencoders, distributed computing, and the integration of linear algebra with deep learning architectures. Applications in quantum computing and autonomous systems further highlight their growing impact.

In summary, PCA and SVD remain indispensable in extracting insight from complex data, and their relevance continues to expand with technological advancements.

Bibliography

[1] Shenai, P. M., Xu, Z., Zhao, Y. (2012). Applications of principal component analysis (PCA) in materials science. Princ. Compon. Anal. Appl, 25-40.

[2] Grané, A., Jach, A. (2014). Applications of principal component analysis (PCA) in food science and technology. Mathematical and statistical methods in food science and technology, 13(1), 57-86.

[3] Ding, S., Zhang, P., Ding, E., Naik, A., Deng, P., Gui, W. (2010). On the application of PCA technique to fault diagnosis. Tsinghua Science and Technology, 15(2), 138-144.

[4] Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, **2**(11), 559–572.

[5] Melzer, T. (2004). SVD and its application to generalized eigenvalue problems. Vienna University of Technology.

[6] Peña, J. M., Sauer, T. (2019). SVD update methods for large matrices and applications. Linear Algebra and its Applications, 561, 41-62.

[7] Biglieri, E., Yao, K. (1989). Some properties of singular value decomposition and their applications to digital signal processing. Signal Processing, 18(3), 277-289.

[8] Akritas, A. G., Malaschonok, G. I. (2004). Applications of singular-value decomposition (SVD). Mathematics and computers in simulation, 67(1-2), 15-31.

[9] Binongo, J. N. G., Smith, M. W. A. (1999). The application of principal component analysis to stylometry. Literary and Linguistic Computing, 14(4), 445-466.

[10] Tzeng, D. Y., Berns, R. S. (2005). A review of principal component analysis and its applications to color technology. Color Research Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur, 30(2), 84-98.

[11] Giuliani, A. (2017). The application of principal component analysis to drug discovery and biomedical data. Drug discovery today, 22(7), 1069-1076.

[12] Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, **24**(6), 417–441.

[13] Golub, G. H., & Kahan, W. (1965). Calculating the singular values and pseudo-inverse of a matrix. *Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis*, **2**(2), 205–224.

[14] Strang, G. (1993). Introduction to Linear Algebra. *Wellesley-Cambridge Press.*

[15] Jolliffe, I. T. (2002). *Principal Component Analysis* (2nd ed.). Springer.

[16] Trefethen, L. N., & Bau, D. (1997). *Numerical Linear Algebra*. SIAM.

[17] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer.

[18] Eckart, C., & Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, **1**, 211–218.

[19] Wall, M. E., Rechtsteiner, A., & Rocha, L. M. (2003). Singular value decomposition and principal component analysis. In *A practical approach to microarray data analysis* (pp. 91–109). Springer.

[20] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[21] Abdi, H., & Williams, L. J. (2010). Principal component analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, **2**(4), 433–459.

[22] Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, **290**(5500), 2323–2326.

[23] Schölkopf, B., Smola, A., & Müller, K. R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, **10**(5), 1299–1319.

[24] Wold, S., Esbensen, K., & Geladi, P. (1987). Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, **2**(1-3), 37–52.

[25] Candès, E. J., Li, X., Ma, Y., & Wright, J. (2009). Robust principal component analysis? *Journal of the ACM*, **58**(3), 11.

[26] Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. *Computer*, **42**(8), 30–37.

[27] Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, **401**(6755), 788–791.

[28] Brand, M. (2002). Incremental singular value decomposition of uncertain data with missing values. In *European Conference on Computer Vision* (pp. 707–720). Springer.

[29] Mahoney, M. W. (2011). Randomized algorithms for matrices and data. *Foundations and Trends in Machine Learning*, **3**(2), 123–224.

[30] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, **521**(7553), 436–444.

[31] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

[32] Narayan, V., & Kundu, A. (2020). Efficient implementation of deep PCA using autoencoders. *IEEE Transactions on Neural Networks and Learning Systems*, **31**(6), 2025–2037.

[33] Preskill, J. (2018). Quantum computing in the NISQ era and beyond. *Quantum*, **2**, 79.

[34] Li, Y., Zhang, L., & Liu, Z. (2019). Neural SVD: Learning structured low-rank representations with neural networks. In *Proceedings of NeurIPS*.

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-42

PLAGIARISM VERIFICATION

Title of the Thesis: Recent Applications of PCA and SVD

Total Pages: 39 Pages   Name of the Scholar: Aditi

Supervisor(s): Mr. Jamkhogam Touthang

Department: Department of Applied Mathematics

This is to report that the above thesis was scanned for similarity detection. Process and outcome is given below:

Software used: Turnitin   Similarity Index: 10%

Total Word Count: 7249 Words

Date: ____________________

Candidate's Signature   Signature of Supervisor

Acceptance of Abstract for 3rd International Conference on Recent Trends in Mathematical Sciences (ICRTMS-2025) Inbox x

ICRTMS2025 27 Apr 2025, 19:26

to me ▾

Dear Aditi Ahuja

I hope you are doing well.

We are pleased to inform you that the Conference Committee reviewed your abstract titled "Recent Applications of PCA and SVD" and has approved for presentation at "3rd **International Conference on Recent Trends in Mathematical Sciences (ICRTMS- 2025)"** scheduled to be held on 10th – 11th **May, 2025** at **Himachal Pradesh University, Shimla**, H. P., India in Hybrid mode.

We believe that your presentation will make a valuable contribution to the conference. Your Paper ID is **ICRTMS_203**

We request you to **fill the registration form,** if not done already, and mail your **full length paper in PDF format** latest by **25th April, 2025.**
Please feel free to contact us for any queries.
To register, please fill out the Google Form available at the link:
https://forms.gle/X1qh8EtQetFBoXLe9

The participants who will not register, will not be allowed to present their paper in the conference.

Lodging Arrangement
The organizing committee of ICRTMS-2025 makes arrangements for the stay of participants in nearby guest houses and hotels. The participants are free to exercise their choice about their stay for which they have to immediately contact the concerned guest house or hotel. **The participants are requested to book their accommodation by the end of March, 2025 as in the months of May and June there is tourist season in Shimla.**

Hotel Green View: Situated at Sangti and is about 1 km from the venue.
Tariff: Rs. 1000/- per person in double or triple sharing room with Balcony and Rs. 750/- per person in double or triple sharing room without Balcony.
Contact Details: +91-98166-87459, +91-70186-26662, +91-78071-86043
Hotel Ganga Palace: Situated at Summer Hill, Shimla and is about 100 meters from the venue.

HIMACHAL GANITA PARISHAD

(REGISTERED UNDER H.P. SOCIETIES REGISTRATION ACT 2006)

3rd INTERNATIONAL CONFERENCE ON RECENT TRENDS IN MATHEMATICAL SCIENCES (ICRTMS-2025)

10th-11th May, 2025

CERTIFICATE OF APPRECIATION

This is to certify that **Ms. Aditi, UG/PG Student, Delhi Technological University** has presented a research paper entitled **Recent Applications of PCA and SVD** in **3rd International Conference on Recent Trends in Mathematical Sciences (ICRTMS-2025)** organized by the **Himachal Ganita Parishad (HGP) at Himachal Pradesh University, Shimla** on **10th-11th May, 2025**.

Dr. Kamalendra Kumar
Co-Convener

Dr. Neetu Dhiman
Convener

Dr. Shalini Gupta
President (HGP)