Principal Component Analysis

A Project Report

Submitted by

Aklanta Borah	BSD-CC-2404
Amartya Amritanshu	BSD-CC-2405
Rajveer Vora	BSD-CC-2416
Somesh Biswas	BSD-CC-2420

30 April 2026

Abstract

Principal Component Analysis (PCA) is a foundational technique in multivariate statistics for reducing the dimensionality of high-dimensional datasets while preserving as much of the original variability as possible. This report presents a self-contained treatment of PCA, covering the motivation arising from the curse of dimensionality, the formal variance-maximisation problem and its solution via the eigenvalue decomposition of the covariance matrix, the geometric interpretation as an orthogonal rotation, and complete worked applications to the Fisher Iris dataset and to the high-dimensional ZIP handwritten-digit dataset (including image reconstruction from a small number of principal components). Diagnostics including the scree plot, proportion of explained variance, and the correlation circle are discussed in detail. The report concludes with a critical appraisal of the method's assumptions and limitations.

Keywords: dimensionality reduction, eigenvalue decomposition, covariance matrix, principal components, variance explained, image reconstruction.

Contents

1. Introduction

Modern statistical datasets routinely consist of a large number of variables measured simultaneously on each experimental unit. While rich in potential information, such high-dimensional data present serious analytical challenges: visualisation becomes impossible beyond three dimensions, many algorithms degrade in accuracy or become computationally prohibitive as p grows, and correlated variables introduce redundancy that obscures the underlying structure.

Dimensionality reduction addresses these challenges by representing each observation in a lower-dimensional space that retains the essential information. Among available techniques, **Principal Component Analysis** (PCA) is the most widely used. sPCA constructs a sequence of new uncorrelated variables the *principal components* as linear combinations of the originals, ordered so that each successive component captures as much of the remaining variance as possible.

The objectives of this report are: (i) to motivate the need for dimensionality reduction; (ii) to derive the PCA solution from first principles; (iii) to describe practical implementation and interpretation; (iv) to demonstrate the method on the Fisher Iris dataset; and (v) to illustrate PCA-based image compression and reconstruction using the ZIP handwritten-digit dataset.

2. The Dimensionality Reduction Problem

2.1. Setup and Notation

Let $\mathbf{X} = (X_1, X_2, \ldots, X_p)^T \in \mathbb{R}^p$ denote a random vector of p measured variables. A dataset of n observations is represented as the $n \times p$ data matrix

$$\mathcal{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix},$$

where row i contains the observation vector $\mathbf{x}_i^T$.

2.2. Motivation

High-dimensional data suffer from three interrelated problems that make direct analysis difficult.

Redundancy. When variables are correlated, multiple columns of $\mathcal{X}$ carry essentially the same information. Retaining all variables therefore wastes computational resources and can destabilise downstream analyses such as regression or clustering.

Noise. Real measurements contain random noise. In high dimensions, a large fraction of the total variance may reside in directions that reflect noise rather than signal. By concentrating on the directions of greatest variance, one can implicitly filter out low-energy noise dimensions.

Computational and statistical inefficiency. Many algorithms scale polynomially or exponentially with p. Furthermore, statistical estimation in high dimensions requires sample sizes that grow with p; reducing the effective dimensionality alleviates this bur-

den.

2.3. Inadequacy of Naive Approaches

Two naive approaches are commonly considered and found wanting. First, *variable deletion* simply discarding some of the p variables risks losing information that is spread across multiple correlated variables and is not recoverable. Second, *equal-weight averaging*, i.e. computing $\bar{X} = p^{-1} \sum_{j=1}^{p} X_j$, treats all variables as equally important and ignores their differential contributions to overall variation. PCA overcomes both deficiencies by constructing an *optimal* weighted linear combination of the variables.

3. Mathematical Derivation of PCA

3.1. The First Principal Component

PCA seeks to project $\mathbf{X}$ onto a one-dimensional subspace spanned by a unit vector $\boldsymbol{\delta} \in \mathbb{R}^p$ such that the variance of the projected scores $Y = \boldsymbol{\delta}^T \mathbf{X}$ is maximised. Formally, the problem is

$$\max_{\boldsymbol{\delta} \in \mathbb{R}^p} \operatorname{Var}(\boldsymbol{\delta}^T \mathbf{X}) \quad \text{subject to} \quad \|\boldsymbol{\delta}\|^2 = \boldsymbol{\delta}^T \boldsymbol{\delta} = 1. \tag{1}$$

Assuming $\mathbb{E}[\mathbf{X}] = \mathbf{0}$ (without loss of generality, by centering), the objective becomes $\operatorname{Var}(\boldsymbol{\delta}^T \mathbf{X}) = \boldsymbol{\delta}^T \boldsymbol{\Sigma} \boldsymbol{\delta}$, where $\boldsymbol{\Sigma} = \operatorname{Var}(\mathbf{X})$ is the $p \times p$ population covariance matrix.

The geometric motivation for maximising variance is illustrated in Figure 1: the total variance of the data is partitioned into the variance of the projection onto a direction $\mathbf{w}$ and the squared reconstruction (orthogonal) error. Since total variance is fixed, minimising the reconstruction error is equivalent to maximising the projected variance.

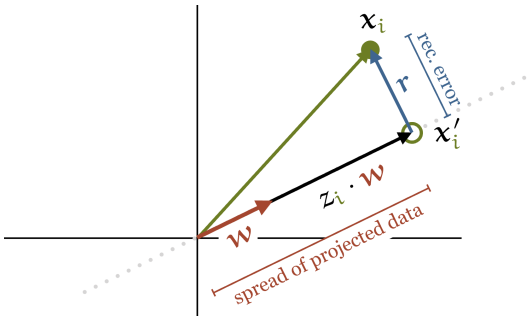


Figure 1: Geometric illustration of the PCA optimisation. For a data point x_i projected onto direction $\mathbf{w}$, the total variance equals the projected variance (spread of $z_i \cdot \mathbf{w}$) plus the squared reconstruction error. Minimising the orthogonal error is therefore equivalent to maximising the projected variance.

3.2. Solution via Lagrange Multipliers

Introducing a Lagrange multiplier λ for the unit-norm constraint, the Lagrangian is

$$\mathcal{L}(\boldsymbol{\delta}, \lambda) = \boldsymbol{\delta}^T \boldsymbol{\Sigma} \boldsymbol{\delta} - \lambda(\boldsymbol{\delta}^T \boldsymbol{\delta} - 1).$$

Setting the partial derivative with respect to $\boldsymbol{\delta}$ to zero yields the necessary condition

$$2\boldsymbol{\Sigma}\boldsymbol{\delta} - 2\lambda\boldsymbol{\delta} = \mathbf{0},$$

which simplifies to the standard eigenvalue equation:

$$\boldsymbol{\Sigma}\boldsymbol{\delta} = \lambda\boldsymbol{\delta}. \tag{2}$$

Thus any maximiser of (1) must be an eigenvector of $\boldsymbol{\Sigma}$. Substituting (2) into the objective confirms that the achieved variance equals the corresponding eigenvalue:

$$\mathrm{Var}(\boldsymbol{\delta}^T\mathbf{X}) = \boldsymbol{\delta}^T\boldsymbol{\Sigma}\boldsymbol{\delta} = \lambda\boldsymbol{\delta}^T\boldsymbol{\delta} = \lambda.$$

To *maximise* variance, one therefore selects the eigenvector $\boldsymbol{\gamma}_1$ associated with the *largest* eigenvalue λ_1. This is the **first principal component direction**, and $Y_1 = \boldsymbol{\gamma}_1^T\mathbf{X}$ is the **first principal component**.

3.3. Subsequent Principal Components

The second principal component $Y_2 = \boldsymbol{\gamma}_2^T\mathbf{X}$ is defined as the unit-norm linear combination of $\mathbf{X}$ with maximum variance subject to being uncorrelated with Y_1. It can be shown that the solution is the eigenvector $\boldsymbol{\gamma}_2$ corresponding to the second largest eigenvalue λ_2. By induction, the j-th principal component direction is the eigenvector of $\boldsymbol{\Sigma}$ corresponding to the j-th largest eigenvalue λ_j.

3.4. The Full PCA Transformation

Since $\boldsymbol{\Sigma}$ is real, symmetric, and positive semi-definite, its eigendecomposition is

$$\boldsymbol{\Sigma} = \boldsymbol{\Gamma}\boldsymbol{\Lambda}\boldsymbol{\Gamma}^T,$$

where $\boldsymbol{\Gamma} = [\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_p]$ is an orthogonal matrix of eigenvectors and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_p)$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$. The full principal component vector is

$$\mathbf{Y} = \boldsymbol{\Gamma}^T(\mathbf{X} - \boldsymbol{\mu}), \tag{3}$$

where $\boldsymbol{\mu} = \mathbb{E}[\mathbf{X}]$. The components $Y_1, \ldots, Y_p$ satisfy the following key properties.

Proposition 1 (Properties of Principal Components)**.** *Let* $\mathbf{Y}$ *be defined by* (3)*. Then:*

(i) $\mathrm{Var}(Y_j) = \lambda_j$ *for each* $j = 1, \ldots, p$.

(ii) $\mathrm{Cov}(Y_i, Y_j) = 0$ *for all* $i \neq j$ *(uncorrelatedness).*

(iii) $\sum_{j=1}^{p} \mathrm{Var}(Y_j) = \sum_{j=1}^{p} \lambda_j = \mathrm{tr}(\boldsymbol{\Sigma}) = \sum_{j=1}^{p} \mathrm{Var}(X_j)$ *(total variance preserved).*

(iv) $\mathrm{Var}(Y_1) \geq \mathrm{Var}(Y_2) \geq \cdots \geq \mathrm{Var}(Y_p)$ *(variance ordering).*

Property (iii) establishes that PCA is a lossless transformation: the total variability in the dataset is preserved, merely redistributed across uncorrelated axes. Property (ii) ensures that the new variables carry no redundant information.

3.5. Geometric Interpretation

Geometrically, PCA performs a rigid rotation of the original coordinate system. The new axes $\boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_p$ are aligned with the principal axes of the data ellipsoid. The intrinsic shape of the point cloud is unchanged; only the coordinate frame from which it is viewed is rotated to align with the directions of maximum spread. Dimensionality reduction is achieved by retaining only the first $q < p$ axes, projecting the data onto the q-dimensional

subspace that best explains total variance in a least-squares sense.

Figure 2 illustrates this geometrically: high-dimensional data are projected onto the two principal axes (PC1 and PC2), which capture the directions of greatest and second-greatest spread respectively.

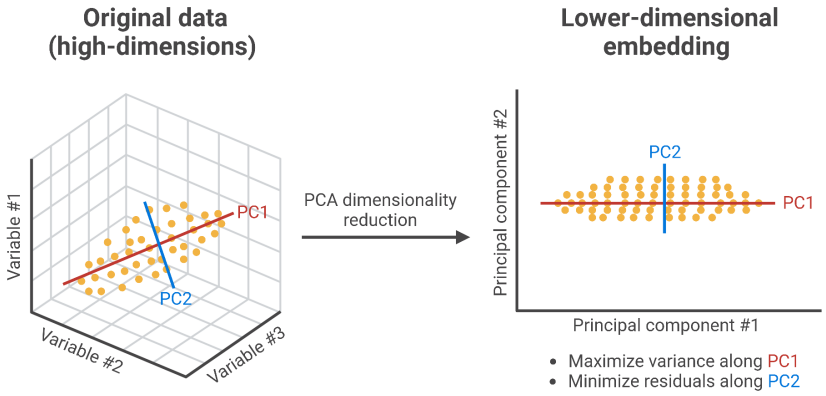


Figure 2: Geometric view of PCA dimensionality reduction. The original data (left, shown in three dimensions) are rotated and projected onto the plane spanned by the first two principal components (right), which align with the directions of maximum and second-maximum variance. The shape of the data cloud is preserved; only the coordinate frame changes.

4. Practical Implementation

4.1. Sample Estimates

In practice, the population PC transformation must be replaced by the respective sample estimators. Given the $n \times p$ data matrix $\mathcal{X}$, the population mean μ becomes the sample mean $\bar{x}$, and the covariance Σ is replaced by the empirical covariance matrix $\mathcal{S}$.

Using the centering matrix $\mathcal{H} = \mathcal{I} - (n-1)^{-1} 1_n 1_n^\top$, we can elegantly express the sample covariance matrix as:

$$\mathcal{S} = (n-1)^{-1} \mathcal{X}^\top \mathcal{H} \mathcal{X} \tag{4}$$

If $\mathcal{S} = \mathcal{G}\mathcal{L}\mathcal{G}^\top$ represents the spectral decomposition of $\mathcal{S}$ (where $\mathcal{G}$ is the matrix of eigenvectors and $\mathcal{L} = \operatorname{diag}(\ell_1, \ldots, \ell_p)$ is the diagonal matrix of eigenvalues), the empirical principal components are obtained by:

$$\mathcal{Y} = (\mathcal{X} - 1_n \bar{x}^\top)\mathcal{G} \tag{5}$$

Furthermore, noting that $\mathcal{H} 1_n \bar{x}^\top = 0$, the covariance matrix of the principal components can be written to demonstrate that the variances equal the eigenvalues:

$$\mathcal{S}_\mathcal{Y} = (n-1)^{-1} \mathcal{Y}^\top \mathcal{H} \mathcal{Y} = (n-1)^{-1} \mathcal{G}^\top \mathcal{X}^\top \mathcal{H} \mathcal{X} \mathcal{G} = \mathcal{G}^\top \mathcal{S} \mathcal{G} = \mathcal{L} \tag{6}$$

4.2. Scale Considerations

PCA is not scale-invariant. If the variables are measured in different units or have very different variances, the variable with the largest variance will dominate the first principal

component regardless of its relevance. In such cases it is standard practice to standardise each variable to unit variance before applying PCA, which is equivalent to performing PCA on the correlation matrix rather than the covariance matrix.

4.3. Choosing the Number of Components

The proportion of total variance explained by the first q components is

$$\psi_q = \frac{\sum_{j=1}^{q} \lambda_j}{\sum_{j=1}^{p} \lambda_j}. \tag{7}$$

Several guidelines exist for selecting q: (i) retaining enough components so that ψ_q exceeds a threshold (commonly 80–95%); (ii) inspecting the *scree plot* of λ_j against j and selecting the point at which the plot bends sharply (the "elbow"); and (iii) the Kaiser rule, which retains components with $\lambda_j > 1$ when PCA is applied to the correlation matrix. No single rule is universally optimal; substantive knowledge should supplement formal criteria.

5. Application to the Fisher Iris Dataset

5.1. Dataset Description

The Fisher Iris dataset comprises $n = 150$ observations of three iris species (*Iris setosa*, *I. versicolor*, and *I. virginica*), each described by four morphometric measurements in centimetres: sepal length (X_1), sepal width (X_2), petal length (X_3), and petal width (X_4). The four variables are all in the same unit, so PCA is applied to the covariance matrix without standardisation.

5.2. Computation

The sample mean vector is $\bar{\mathbf{x}} = [5.843,\ 3.057,\ 3.758,\ 1.199]$. Following centering and covariance matrix estimation, eigendecomposition yields the loadings and variance summary shown below. The PCA scores may be computed either manually from the covariance matrix or directly using built-in software.

Table 1: Eigenvector loadings for the first two principal components.

Variable	PC1	PC2
X_1 Sepal Length	+0.361	+0.657
X_2 Sepal Width	−0.085	+0.730
X_3 Petal Length	+0.857	−0.173
X_4 Petal Width	+0.358	−0.075

Table 2: Variance explained by each component.

Component	Var.(%)	Cum.(%)
PC1	92	92
PC2	5	97
PC3	2	99
PC4	1	100

Manual Computation in R

```
data(iris)
X  <- as.matrix(iris[,1:4])
mu <- colMeans(X)
Xc <- sweep(X, 2, mu)

S   <- t(Xc) %*% Xc / nrow(X)-1
eig <- eigen(S)
Y   <- Xc %*% eig$vectors
Y_2 <- Y[,1:2]
```

Built-in PCA in R

```
pca <- prcomp(X,
       center = TRUE,
       scale. = FALSE)

Y_2_prcomp <-
pca$x[,1:2]
```

Both approaches produce the same two-dimensional principal component scores, up to possible sign changes in the eigenvectors.

5.3. Interpretation of Loadings

The first principal component ($y_1 = 0.361x_1 - 0.085x_2 + 0.857x_3 + 0.358x_4$) is dominated by petal length and, to a lesser extent, sepal length and petal width. It may be interpreted as an index of overall flower size, with petal dimensions contributing most strongly. The second principal component ($y_2 = 0.657x_1 + 0.730x_2 - 0.173x_3 - 0.075x_4$) is driven primarily by sepal length and sepal width, capturing variation in sepal shape that is orthogonal to the overall size axis.

5.4. Dimensionality Reduction and Visualisation

As shown in Table 2, the first two components jointly explain 97% of the total variance. The 150×4 data matrix is therefore well approximated by the 150×2 score matrix $\mathcal{Y}_{(2)}$. When the 150 score vectors are plotted in the (y_1, y_2) plane (Figure 3), the three species form well-separated clusters: *I. setosa* occupies a compact region at strongly negative y_1 values, while *I. versicolor* and *I. virginica* are separated along both axes. This clear species discrimination not imposed by the method but emerging naturally from the data validates the effectiveness of the two-component solution.

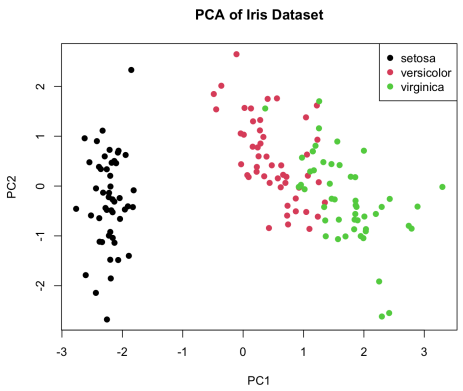


Figure 3: PCA biplot of the Fisher Iris dataset projected onto the first two principal components. The three species (*I. setosa*, *I. versicolor*, *I. virginica*) separate clearly in the two-dimensional PC space, despite no class labels being used in the analysis. PC1 accounts for 92% of total variance and PC2 for a further 5%.

5.5. Scree Plot

Figure 4 shows the scree plot for the Iris dataset. The sharp elbow after the first component, which alone explains 92% of total variance, confirms that a one or two component solution is sufficient. The cumulative curve reaches 97% at $q = 2$ and 99% at $q = 3$, leaving only negligible residual variation in the fourth component.

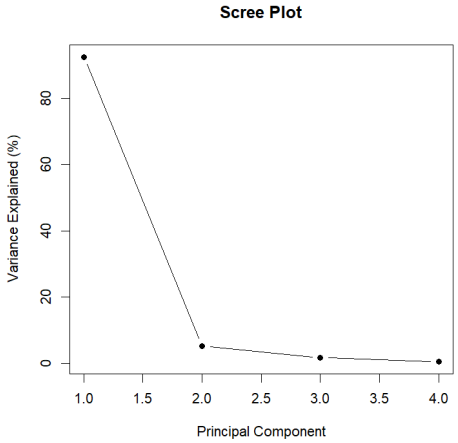


Figure 4: Scree plot for the Fisher Iris dataset showing the percentage of total variance explained by each principal component.

5.6. Correlation Circle

The correlation between the i-th original variable and the j-th PC is

$$r_{X_i Y_j} = g_{ij} \left(\frac{l_j}{s_{X_i X_i}} \right)^{1/2},$$

where g_{ij} is the (i, j) element of G, l_j is the j-th eigenvalue, and $s_{X_i X_i}$ is the sample variance of X_i. Plotting $(r_{X_i Y_1},\ r_{X_i Y_2})$ for each variable yields the *correlation circle* (Figure 5). Since $r^2_{X_i Y_1} + r^2_{X_i Y_2} \leq 1$, all points fall within the unit disc; variables near the boundary are well represented by the two-component solution. In the Iris correlation circle, X_1, X_3, and X_4 cluster near the right boundary with high positive $r_{X_i Y_1}$, confirming their dominant role in PC1. Variable X_2 lies near the top boundary with high $r_{X_2 Y_2}$, confirming its role in PC2.

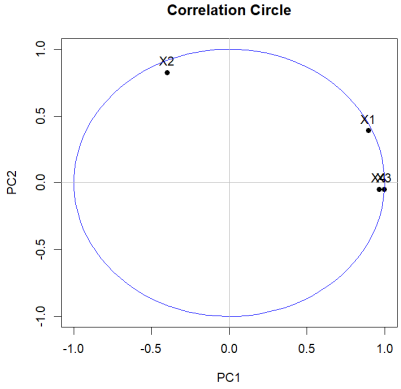


Figure 5: Correlation circle for the Fisher Iris dataset. Each point represents the correlation of an original variable with PC1 (horizontal axis) and PC2 (vertical axis).

6. Application to the ZIP Handwritten-Digit Dataset: Image Compression and Reconstruction

A compelling illustration of PCA's power is its ability to compress and reconstruct high-dimensional data in this case, images using only a small fraction of the original features. We apply PCA to a subset of the ZIP handwritten-digit dataset [4], focusing on 650 images of the digit 3.

6.1. Dataset Description

Each image is a 16×16 greyscale pixel grid, which is vectorised into a single observation $\mathbf{x}_i \in \mathbb{R}^{256}$. The full dataset is therefore represented as a 650×256 data matrix where each entry x_{ij} records the intensity of pixel j in image i. With $p = 256$ features, direct visualisation or analysis of the full data matrix is infeasible.

6.2. PCA Transformation

Let $\mathcal{X} \in \mathbb{R}^{650\times256}$ denote the data matrix of handwritten digit images, where each row is a vectorised 16×16 image. Let $\bar{\mathbf{x}} \in \mathbb{R}^{256}$ be the sample mean pixel vector. The centering, covariance matrix, and spectral decomposition are given by

$$X_c = \mathcal{X} - \mathbf{1}_n \bar{\mathbf{x}}^T, \qquad S = \frac{1}{n-1} X_c^T X_c, \qquad S = G \Lambda G^T.$$

where $G = [g_1, g_2, \ldots, g_{256}]$ is the orthogonal matrix of eigenvectors and

$$\Lambda = \operatorname{diag}(\ell_1, \ell_2, \ldots, \ell_{256}), \qquad \ell_1 \geq \ell_2 \geq \cdots \geq \ell_{256} \geq 0.$$

Projecting the centered data onto this basis gives the principal component score matrix

$$Y = X_c G, \tag{8}$$

where $Y \in \mathbb{R}^{650\times256}$. Each row of Y contains the principal component coordinates of one image. Since G is orthogonal, the centered data may be recovered exactly by

$$X_c = Y G^T.$$

Thus PCA rewrites the original pixel data in a new coordinate system whose axes are ordered according to decreasing variance explained.

6.3. Low-Rank Approximation and Reconstruction

From the PCA transformation,

$$Y = X_c G,$$

where $X_c = \mathcal{X} - \mathbf{1}_n \bar{\mathbf{x}}^T$ and G is the orthogonal matrix of eigenvectors. Since $G^{-1} = G^T$, we obtain

$$X_c = Y G^T.$$

Hence the original data matrix can be written as

$$\mathcal{X} = \mathbf{1}_n \bar{\mathbf{x}}^T + YG^T.$$

To compress the data, retain only the first $q \ll 256$ principal components. Let

$$G_{(q)} = [g_1, \ldots, g_q], \qquad Y_{(q)} = X_c G_{(q)}.$$

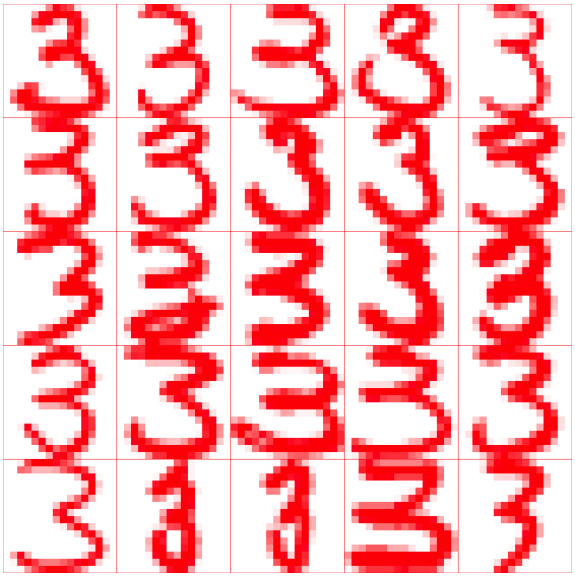
Replacing the full score matrix by $Y_{(q)}$ gives the rank-q reconstruction:

$$\boxed{\hat{\mathcal{X}}_{(q)} = \mathbf{1}_n \bar{\mathbf{x}}^T + Y_{(q)} G_{(q)}^T} \tag{9}$$

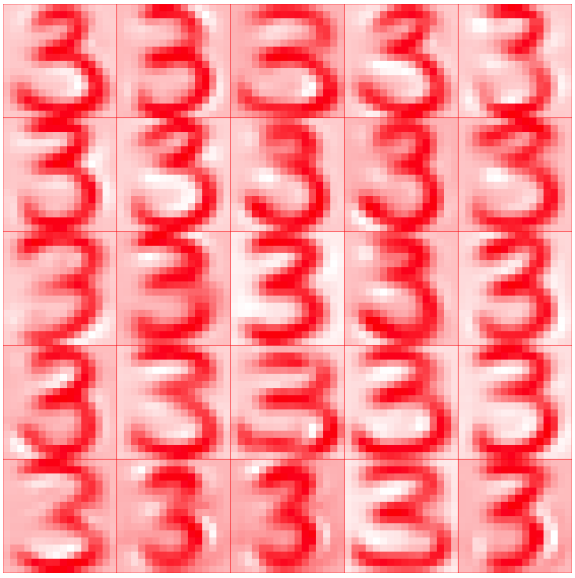
Thus each image is represented using only q scores instead of 256 pixel values, then reshaped back to a 16×16 grid.

6.4. Visualisation

Figure 6 illustrates the visual quality of reconstruction at $q = 10$ principal components. The reconstructed images are noticeably smoother than the originals because the fine pixel-level variation captured by components 11–256 is discarded, effectively denoising the images. Despite retaining only 3.9% of the original features ($10/256 \approx 0.039$), the digit "3" remains clearly legible in every reconstructed image.



(a) Original handwritten "3" images from the ZIP dataset (16×16 pixels each).



(b) Reconstructions using only the first $q = 10$ principal components (3.9% of original features).

Figure 6: PCA-based image reconstruction of handwritten digit "3" from the ZIP dataset.Despite retaining only 10 out of 256 pixel dimensions, the digit remains clearly legible in every reconstructed image, though fine-grained stroke detail is smoothed out.

6.5. Reconstruction Error

For the i-th image, the discarded components are $y_{i,q+1}, \ldots, y_{i,256}$. Since the eigenvectors are orthonormal, the squared reconstruction error is

$$\|\mathbf{x}_i - \hat{\mathbf{x}}_{i,(q)}\|^2 = \sum_{j=q+1}^{256} y_{ij}^2.$$

Summing over all $n = 650$ images and using

$$\ell_j = \frac{1}{n-1} \sum_{i=1}^{n} y_{ij}^2,$$

we obtain

$$\sum_{i=1}^{650} \|\mathbf{x}_i - \hat{\mathbf{x}}_{i,(q)}\|^2 = (n-1) \sum_{j=q+1}^{256} \ell_j.$$

Hence the total reconstruction error equals the variance discarded by omitting the remaining components. Reversing the order of summation and substituting $\sum_{i=1}^{n} y_{ij}^2 = (n-1)\ell_j$ confirms this directly:

$$\boxed{\sum_{i=1}^{650} \|\mathbf{x}_i - \hat{\mathbf{x}}_{i,(q)}\|^2 = (n-1) \sum_{j=q+1}^{256} \ell_j.}$$

Therefore, minimising reconstruction error is equivalent to maximising retained variance, which is the central principle of Principal Component Analysis.

Table 3 reports the three error metrics at representative values of q, computed directly from the formula above. The mean SSE per image is the total SSE divided by n, and the mean MSE per pixel divides further by $p = 256$.

Table 3: Reconstruction error at selected q for the digit-3 dataset ($n = 650$, $p = 256$).

q (PCs)	Total SSE	Mean SSE / image	Mean MSE / pixel	Var. retained (%)
1	51726.75	78.6121	0.307078	12.7
5	34175.01	51.9377	0.202882	42.3
10	24182.66	36.7518	0.143562	59.2
20	15223.66	23.1363	0.090376	74.3
50	6083.57	9.2455	0.036115	89.7
100	1888.75	2.8704	0.011213	96.8
150	569.84	0.8660	0.003383	99.0
200	105.70	0.1606	0.000628	99.8
256	0.00	0.0000	0.000000	100.0

The error falls sharply in the first few components: at $q = 10$ the total SSE is already less than half its value at $q = 1$, while the mean MSE per pixel is only 0.144. Beyond $q = 50$ the gains diminish markedly, with $q = 150$ components sufficing to reduce the mean per-image error below 1.

6.6. Variance Explained and Compression Ratio

The proportion of total pixel variance captured by q components is given by (7). Table 4 reports representative values for the handwritten-threes dataset.

Table 4: Cumulative proportion of variance explained by the leading principal components for the ZIP handwritten-threes dataset ($p = 256$).

q (components retained)	Variance Explained (%)	Compression ratio (p/q)
1	≈ 12.7	256 : 1
5	≈ 42.3	256 : 5
10	≈ 59.2	256 : 10
20	≈ 74.3	256 : 20
50	≈ 89.7	256 : 50

With $q = 10$ components, the reconstruction captures roughly 59.2% of the total pixel variance while storing only 10 scores instead of 256 pixel values, giving a compression ratio of approximately 256 : 10. The main idea behind taking $q = 10$ was to compress the data as much as with being able to retain the approximate original images. For a better variance we can take $q = 50$ capturing about 89.7% of the total variance with compression ratio 256 : 50 which would also be an ideal situation.

6.7. R Implementation

The entire pipeline can be executed concisely in R:

```
# Assume X is the 650 x 256 matrix of pixel values
n   <- nrow(X)
mu  <- colMeans(X)
Xc  <- sweep(X, 2, mu)                  # Step 1: center
S   <- t(Xc) %*% Xc / (n - 1)          # Step 2: covariance
eig <- eigen(S)                          # Step 3: eigendecompose
G   <- eig$vectors                       # eigenvectors (256 x 256)
Y   <- Xc %*% G                          # Step 4: transform

# Reconstruct with q = 10 components
q        <- 10
G_q      <- G[, 1:q]
Y_q      <- Y[, 1:q]
X_hat_q  <- sweep(Y_q %*% t(G_q), 2, -mu)  # add back mean

# Reshape row i to a 16x16 image:  matrix(X_hat_q[i, ], 16, 16)
```

7. Discussion and Limitations

PCA is a powerful tool for exploratory analysis and preprocessing, with strong theoretical guarantees: optimality of the linear approximation, preservation of total variance, and uncorrelatedness of components. The image reconstruction example further illustrates that variance maximisation and reconstruction error minimisation are two faces of the same

optimisation. Several important limitations must nonetheless be acknowledged.

Linearity. PCA constructs only linear combinations of the original variables and fails to capture nonlinear manifold structure. Kernel PCA, t-SNE, UMAP, and autoencoders extend the idea to nonlinear settings.

Interpretability. Principal components are mathematical constructs and rarely have direct domain interpretations, as each is a weighted combination of all variables. Sparse PCA constrains the loadings to be sparse, improving interpretability at the cost of strict optimality.

Sensitivity to outliers. The sample covariance matrix is non-robust; a few extreme observations can disproportionately influence the principal directions. Robust variants based on the minimum covariance determinant estimator are available.

Choice of q. No universally optimal rule exists for selecting the number of retained components. Formal criteria (scree plot elbow, cumulative variance threshold, Kaiser rule) should be supplemented by substantive domain knowledge.

Stationarity. PCA assumes a constant covariance structure across the dataset. In time-series or spatially varying data this assumption may be violated; dynamic PCA and localised variants address this issue.

8. Conclusion

This report has presented a rigorous treatment of Principal Component Analysis, from its motivation in the challenges of high-dimensional data through its mathematical derivation and practical implementation. The core insight is that PCA identifies the coordinate system in which the data exhibit the greatest variance, providing a compact and informative low-dimensional representation. Applied to the Fisher Iris dataset, a reduction from four to two dimensions retains 97% of the total variance and reveals a clear separation among the three species. Applied to the ZIP handwritten-digit dataset, a reduction from 256 pixel dimensions to just 10 principal components retaining only 3.9% of the original features produces visually recognisable image reconstructions via the formula $\hat{\mathcal{X}}_{(q)} = \mathbf{1}_n \bar{\mathbf{x}}^T + Y_{(q)} G_{(q)}^T$. Despite its limitations linearity, sensitivity to outliers, and interpretability challenges PCA remains an indispensable first step in the analysis of multivariate data.

References

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

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

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

[4] James, G., Witten, D., Hastie, T., and Tibshirani, R. (2023). *An Introduction to Statistical Learning with Applications in R* (2nd ed.). Springer, New York.

[5] Fisher, R. A. (1936). *The use of multiple measurements in taxonomic problems* (Iris dataset).

[6] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning Data Sets: Zip Code Digits Data*. Available at: `https://hastie.su.domains/ElemStatLearn/data.html`.

[7] Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning* (2nd ed.). Springer, New York.