Project “Topic: Eigenvalues and Eigenvectors in Principal Component Analysis”
Title
IB3702 Mathematics for Machine Learning

Harman Singh

15 November, 2025

1 Introduction

Machine learning (ML) relies heavily on mathematics. Luckily, ML is a rather new branch of computer science, which means that it has access to hundreds of years of advancements in the field of mathematics to base its core around, rather than having to discover new mathematical concepts. This means, usually, that the mathematics used in ML is rather straightforward. You build upon centuries worth of linear algebra and calculus to model, analyse and interpret data.

One technique to interpret data, and the topic of this report, is ‘Principal Component Analysis’ (PCA). PCA is a linear dimensionality reduction technique used in exploratory data analysis, with one main purpose — to reduce the dimensionality of a dataset, i.e., to reduce the number of columns (variables) which in turn makes the underlying dataset easier to process by computers. It does this by transforming the data into a new set of variables, the principal components (PCs), which are uncorrelated and ordered by the amount of variance they capture from the original data. The first principal component captures the most variance, the second captures the second most, and so on. This transformation is achieved through the mathematical concepts of eigenvalues and eigenvectors.

The aim of this report is to explore a new field of mathematics and gain knowledge in it. We will connect the theoretical concepts of eigenvalues and eigenvectors from linear algebra to their practical use case in PCA.

Contents

2 Preliminaries

2.1 Vectors

You can visualise a column in a dataset as a vector in a high-dimensional space. Each row in the dataset corresponds to a component of the vector, so a dataset with n observations can be represented as a vector in an n-dimensional space.

x	y
0.41	0.36
0.24	0.09
0.77	0.66

$$\rightarrow \quad \mathbf{v_x} = \begin{bmatrix} 0.41 \\ 0.24 \\ 0.77 \end{bmatrix}, \quad \mathbf{v_y} = \begin{bmatrix} 0.36 \\ 0.09 \\ 0.66 \end{bmatrix}$$

Here, the columns $\mathtt{x}$ and $\mathtt{y}$ from the dataset are represented as vectors $\mathbf{v_x}$ and $\mathbf{v_y}$ in a 3-dimensional space.

2.2 Linear transformations

A linear transformation takes a vector as input and produces another vector as output, while maintaining the structure of the vector space. For example, a linear transformation is represented by a matrix A:

$$A = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}$$

When this matrix transforms the vector $\mathbf{v} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, the result is:

$$A\mathbf{v} = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

This transformation scales the first component of the vector by 2 and the second component by 3.

2.3 Eigenvectors and Eigenvalues

An eigenvector is a special type of vector where, when a linear transformation is applied to it, only the scale of the vector changes but not its direction. The eigenvalue is the factor by which the eigenvector scaled.

It is easier explained with a visualiation. In the figure below, the vector v retained the same direction after a linear transformation, unlike the vector w. This means that v is an eigenvector and the distance between v and Av is its eigenvalue

The matrix and vectors used in the figure are:

$$A = \begin{bmatrix} 2 & 2 \\ -4 & 8 \end{bmatrix}, \quad v = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad w = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

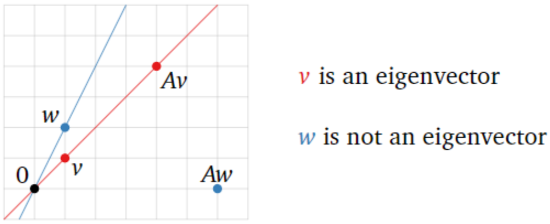

Figure 1: Visualization of eigenvectors and eigenvalues under a linear transformation [1].

If we calculate Av and Aw we can confirm that the vector v was simply scaled by a factor of 4, and that w changed direction.

$$Av = \begin{bmatrix} 2 & 2 \\ -4 & 8 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix} = 4 \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$Aw = \begin{bmatrix} 2 & 2 \\ -4 & 8 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 0 \end{bmatrix}$$

2.4 Eigen-decomposition

Eigen-decomposition is a method used to break down a square matrix into its eigenvalues and eigenvectors. For a square matrix A, the eigen-decomposition can be expressed as:

$$A = V\Lambda V^{-1}$$

where V is a matrix whose columns are the eigenvectors of A, and Λ is a diagonal matrix containing the corresponding eigenvalues.

We take the example matrix A from the previous section:

$$A = \begin{bmatrix} 2 & 2 \\ -4 & 8 \end{bmatrix}$$

For $\lambda = 4$

$$(A - 4I)v = 0 \implies \begin{bmatrix} -2 & 2 \\ -4 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0 \implies -2x + 2y = 0 \implies y = x$$

For $\lambda = 6$

$$(A - 6I)v = 0 \implies \begin{bmatrix} -4 & 2 \\ -4 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 0 \implies -4x + 2y = 0 \implies y = 2x$$

So the eigenvectors corresponding to the eigenvalues $\lambda_1 = 4$ and $\lambda_2 = 6$ are:

$$v_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad v_2 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

2.5 Covariance matrix

Previously we explained that PCA transforms data into a new set of variables, the principal components, which are uncorrelated and ordered by the amount of variance they capture from the original data. To find these principal components, PCA uses the eigen-decomposition of the covariance matrix (a measure of how variables in a dataset vary together).

The covariance matrix S is defined as:

$$S = \frac{1}{n-1}(X - \bar{X})^T (X - \bar{X})$$

where X is the data matrix and $\bar{X}$ is the mean vector.

3 Methods

3.1 The PCA algorithm

3.1.1 Standardise the dataset

Subtract the mean and scale by the standard deviation for each feature to make sure that each feature contributes equally. For this we use the z-score transformation formula, but adapted for each feature j:

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$$

with μ_j being the mean of feature j and σ_j being the standard deviation of feature j.

3.1.2 Calculate the covariance matrix

The covariance matrix S is calculated as:

$$S = \frac{1}{n-1} Z^T Z$$

with Z being the standardised data matrix.

3.1.3 Perform eigen-decomposition on the covariance matrix

We perform eigen-decomposition on the covariance matrix S:

$$S = V \Lambda V^{-1}$$

with V being the matrix of eigenvectors and Λ is the diagonal matrix of eigenvalues.

3.1.4 Sort eigenvalues and select top k eigenvectors

$$\boxed{\lambda_1 \geq \lambda_2} \geq \lambda_3 \geq \cdots \geq \lambda_p$$

PCA is not a lossless algorithm, meaning that we do lose some information when we reduce the dimensionality of the data. However, by selecting the top k eigenvectors (principal components), we retain the directions that capture the most variance in the data, and this usually gives a pretty good approximation of the original data.

4 Numerical Examples

5 Collaboration

6 Reflection

6.1 Student a: Tobias Hungwe

Text...

6.2 Student b: Harman Singh

References

[1] Libretexts. 8.5.3: Eigenvalues and eigenvectors - visualizations, August 2023.