Machine Learning – 04

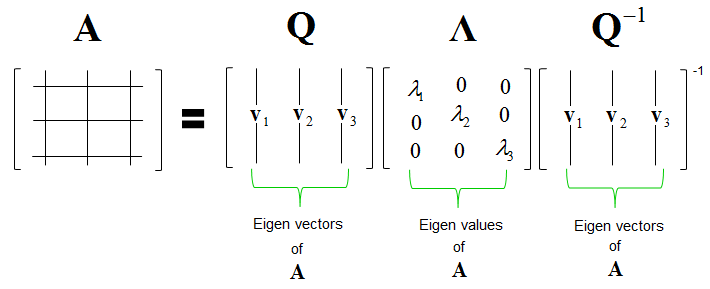
**Eigendecomposition of a matrix**

The Eigendecomposition of a square matrix is a factorization s.t the matrix is represented by its **eigenvalues and eigenvectors** (this concept is often called diagonalization of a matrix)

Let **A** be a square *n* × *n* matrix with *n* linearly independent eigenvectors *qi* (where *i* = 1, ..., *n*). Then **A** can be [factorized](https://en.wikipedia.org/wiki/Matrix_decomposition) as:

**A = Q Λ Q^-1**

�=���−1where **Q** is the square *n* × *n* matrix whose *i*th column is the **eigenvector** *qi* of **A**, and **Λ** is the [diagonal matrix](https://en.wikipedia.org/wiki/Diagonal_matrix) whose diagonal elements are the corresponding **eigenvalues**, *Λii* = *λi*

Note that only [diagonalizable matrices](https://en.wikipedia.org/wiki/Diagonalizable_matrix) can be factorized in this way, trivially

**A picture containing text

Description automatically generated**

**Remark:**

* The eigenvalues of a matrix can be found by **det(A - Iλ) = 0**
  + Where I is the n x n identity matrix
* The eigenvectors of a matrix can be found by substituting for each eigenvalue in the spectrum and computing **Ker(A-Iλ)**

Graphical user interface, text, application

Description automatically generated

Link : https://juanitorduz.github.io/the-spectral-theorem-for-matrices/

**Principal component analysis (PCA)**

PCA is used to reduce the dimensionality of a dataset (see overfitting – lack of generalization), using a transformation that preserves the most variance in the data using **the least amount of dimensions**

It in fact is assumed that the information is carried in the **variance** of the features, meaning that:

**Higher variation ⇒ more information**

We aim to follow these steps:

1. Construct the **covariance matrix** (see previous document)
2. Compute its eigenvalues
3. Use the eigenvectors to reconstruct data

This is useful in **unsupervised learning**