Principal Component Analysis and Linear Discriminant Analysis

Joris Edelmann
joris.edelmann@ovgu.de

21st of August 2025

Table of Contents

Aims and Ideas

Aim: Classify data

Example: We measured 200 spectra of different materials. We want to now, which material contains which ingredients.

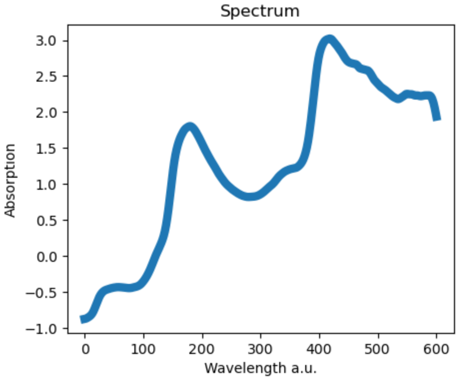


Figure: Example of measured spectra

Source: nirpyresearch

Aim: Classify data

Example: We measured 200 spectra of different materials. We want to now, which material contains which ingredients.

Strategy:

- reduce dimensions and get rid of noise

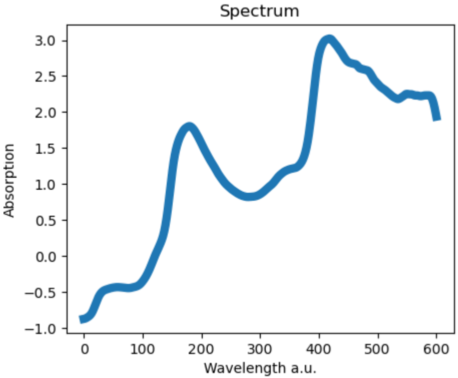


Figure: Example of measured spectra

Source: nirpyresearch

Aim: Classify data

Example: We measured 200 spectra of different materials. We want to now, which material contains which ingredients.

Strategy:

- reduce dimensions and get rid of noise
- optimize distances
- only linear transformations

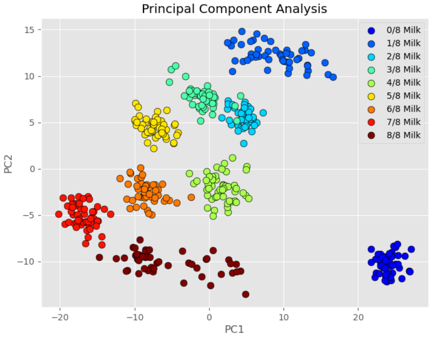


Figure: Reduced to two dimensions

Source: nirpyresearch

Aim: Classify data

Example: We measured 200 spectra of different materials. We want to now, which material contains which ingredients.

Strategy:

- reduce dimensions and get rid of noise
- optimize distances
- only linear transformations
- criteria for classification

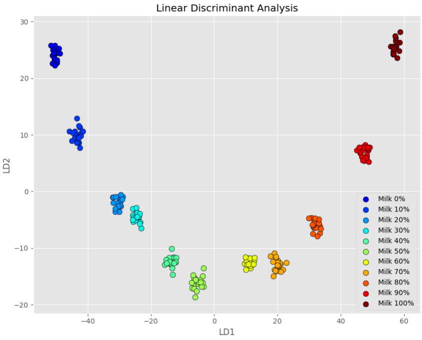
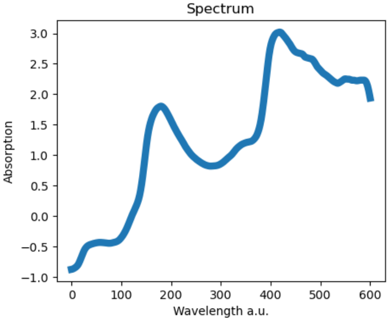


Figure: Easier to decide between classes

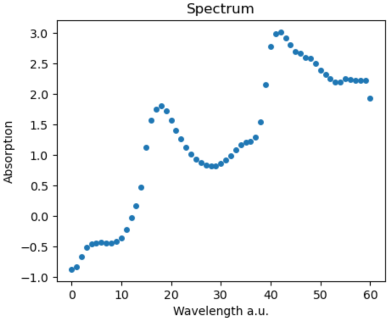
Source: nirpyresearch

Principal Component Analysis

Observations



Observations

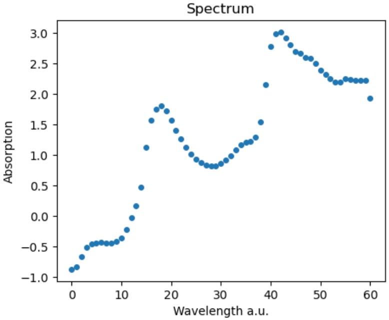


Typical spectrum

- high dimensional (every observed wavelength is new dimension)
- highly correlated (neighboring wavelengths have similar intensity)
- noisy
- few characteristic peaks

Decompose into few spectra (= change of basis, e.g. peaks)

Observations



Typical spectrum

- high dimensional (every observed wavelength is new dimension)
- highly correlated (neighboring wavelengths have similar intensity)
- noisy
- few characteristic peaks

Decompose into few spectra (= change of basis, e.g. peaks)

Prerequisites for PCA

- Standardize (0 mean and unit variance)
- Correlated data

Mathematical formulation

Let v^1, v^2 be two instances (e.g. spectra) and *mean-free*. Recall, the correlation of both vectors is given by the scalar product:

$$\langle v^1, v^2 \rangle = \sum_{i=1}^{n} v_i^1 v_i^2 \tag{1}$$

Input: m instances with n attributes each
See the data as an $m \times n$-matrix A
The covariance matrix is

$$Cov(A, A) = A^T \cdot A = \begin{pmatrix} \langle v^1, v^1 \rangle & \langle v^1, v^2 \rangle & \dots & \langle v^1, v^n \rangle \\ \langle v^2, v^1 \rangle & \langle v^2, v^2 \rangle & \dots & \langle v^2, v^n \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle v^n, v^1 \rangle & \langle v^n, v^2 \rangle & \dots & \langle v^n, v^n \rangle \end{pmatrix} \tag{2}$$

Mathematical formulation II

$$Cov(A,A) = A^T \cdot A = \begin{pmatrix} \langle v^1, v^1 \rangle & \langle v^1, v^2 \rangle & \dots & \langle v^1, v^n \rangle \\ \langle v^2, v^1 \rangle & \langle v^2, v^2 \rangle & \dots & \langle v^2, v^n \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle v^n, v^1 \rangle & \langle v^n, v^2 \rangle & \dots & \langle v^n, v^n \rangle \end{pmatrix} \tag{3}$$

- Compute Eigenvalues and Eigenvectors of $A^T A$ (or SVD from A)
- Eigenvectors are called principal components (PCs)
- Eigenvalues are the relevance of the PC

Usage: Dimensional reduction

Take only take first few PCs

- contain most of the information
- PCs are linear independent and uncorrelated
- first Eigenvalues are easy to compute (Power Method, QR)
- unimportant PCs contain mostly noise

Usage: Dimensional reduction

Take only take first few PCs

- contain most of the information
- PCs are linear independent and uncorrelated
- first Eigenvalues are easy to compute (Power Method, QR)
- unimportant PCs contain mostly noise

Unsupervised learning

Example

1. Standardize data for every wavelength(mean 0, variance 1)
2. Calculate Eigenvalues and Eigenvectors of $A^T \cdot A$

Example

1. Standardize data for every wavelength(mean 0, variance 1)
2. Calculate Eigenvalues and Eigenvectors of $A^T \cdot A$
3. Look at variance ratio (take up to 80%) e.g. PC1: 70%, PC2: 10%, PC3: 6%, PC4: 2%,...
4. Plot PCs against each other

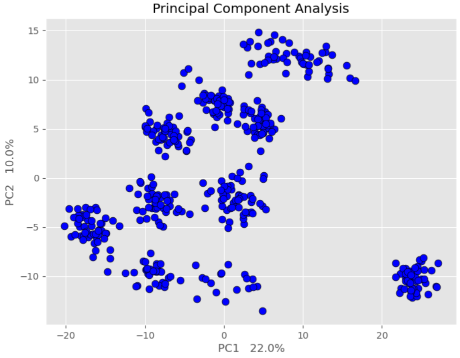


Figure: PC1 vs PC2

Example

1. Standardize data for every wavelength(mean 0, variance 1)
2. Calculate Eigenvalues and Eigenvectors of $A^T \cdot A$
3. Look at variance ratio (take up to 80%) e.g. PC1: 70%, PC2: 10%, PC3: 6%, PC4: 2%,...
4. Plot PCs against each other

Maybe additional data can help.

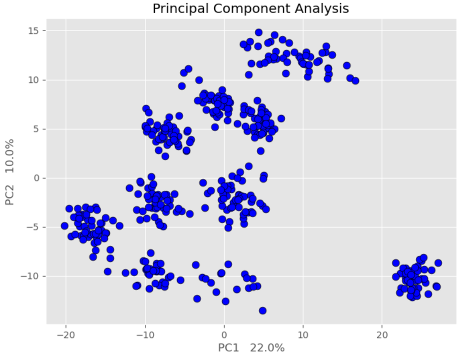


Figure: PC1 vs PC2

Linear Discriminant Analysis

Overview LDA

Linear Discriminant Analysis (**LDA**):

- classifier (sorts unknown data into known classes)
- LDA is a supervised method
- maximizes between-class variance
- minimizes within-class variance
- finds a decision boundary

Prerequisites for LDA

- data is linearly independent Gaussian distributed

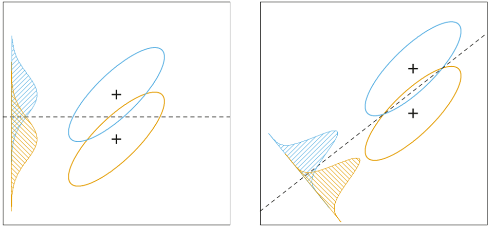


Figure: Idea of LDA

Mathematical formulation

Given N instances $v_i \in \mathbb{R}^d$, separated into K classes $C_1, \ldots, C_K$. Define mean vector for each class C_i, $m_i = \frac{1}{|C_i|} \sum_{v \in C_i} v \in \mathbb{R}^d$ and the overall mean, $m = \frac{1}{N} \sum_{i=0}^{N} v_i \in \mathbb{R}^d$

- Between-class variance (to maximize):

$$S_B = \sum_{i=1}^{K} |C_i|(m_i - m)(m_i - m)^T \in \mathbb{R}^{d \times d}$$

- Within-class variance (to minimize):

$$S_W = \sum_{k=1}^{K} \sum_{x_i \in C_k} (x_i - m_k)(x_i - m_k)^T \in \mathbb{R}^{d \times d}$$

Mathematical formulation II

$$S_B = \sum_{i=1}^{K} |C_i|(m_i - m)(m_i - m)^T \qquad S_W = \sum_{k=1}^{k} \sum_{x_i \in C_k} (x_i - m_k)(x_i - m_k)^T \tag{4}$$

Find transformation $W : \mathbb{R}^d \to \mathbb{R}^l$ $(l \leq d)$ to maximize

$$\mathcal{L}(W, \lambda) = W^T S_B W - \lambda(W^T S_W W - 1) \tag{5}$$

Mathematical formulation II

$$S_B = \sum_{i=1}^{K} |C_i|(m_i - m)(m_i - m)^T \qquad S_W = \sum_{k=1}^{k} \sum_{x_i \in C_k} (x_i - m_k)(x_i - m_k)^T \tag{4}$$

Find transformation $W : \mathbb{R}^d \to \mathbb{R}^l \, (l \leq d)$ to maximize

$$\mathcal{L}(W, \lambda) = W^T S_B W - \lambda(W^T S_W W - 1) \tag{5}$$

$$\frac{\partial \mathcal{L}}{\partial W} = 2S_B W - 2\lambda S_W W = 0$$

Mathematical formulation II

$$S_B = \sum_{i=1}^{K} |C_i|(m_i - m)(m_i - m)^T \qquad S_W = \sum_{k=1}^{k} \sum_{x_i \in C_k} (x_i - m_k)(x_i - m_k)^T \tag{4}$$

Find transformation $W : \mathbb{R}^d \to \mathbb{R}^l \, (l \leq d)$ to maximize

$$\mathcal{L}(W, \lambda) = W^T S_B W - \lambda(W^T S_W W - 1) \tag{5}$$

$$\frac{\partial \mathcal{L}}{\partial W} = 2S_B W - 2\lambda S_W W = 0 \quad \Rightarrow \quad S_B W = \lambda S_W W \tag{6}$$

Decision Boundary

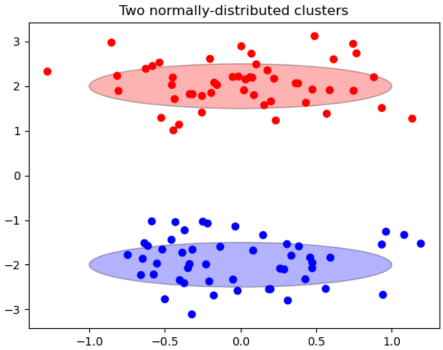


Figure: Where to place decision boundary?

Decision Boundary

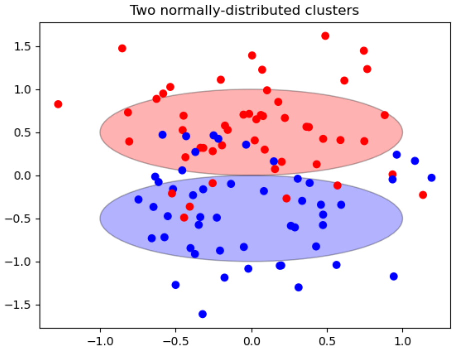


Figure: Where to place decision boundary?

Decision Boundary

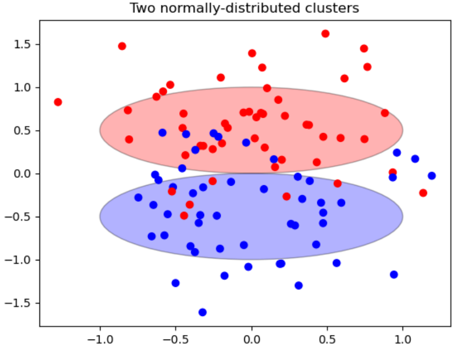


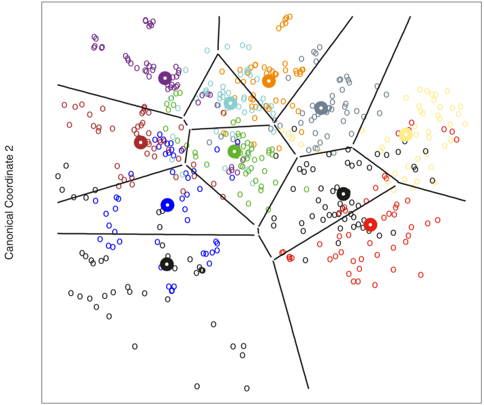
Figure: Where to place decision boundary?

- W gives a new coordinate system
- set decision boundary in 2d, s.t.

$$\frac{\#Upper}{\#Lower} = \frac{|C_1|}{|C_2|}$$

Decision Boundary

Classification in Reduced Subspace



- W gives a new coordinate system
- set decision boundary in 2d, s.t.

$$\frac{\#Upper}{\#Lower} = \frac{|C_1|}{|C_2|}$$

- Do this in 2d successively,

PCA vs LDA

Comparison PCA vs LDA

	PCA	**LDA**
Input	correlated	linearly independent
Output	linearly independent	classes
Purpose	dimensional reduction	classification
learning	unsupervised	supervised

Comparison PCA vs LDA

	PCA	LDA
Input	correlated	linearly independent
Output	linearly independent	classes
Purpose	dimensional reduction	classification
learning	unsupervised	supervised

similar applications because

- classes and PCs are correlated
- LDA works with weakly correlated data too

Comparison PCA vs LDA

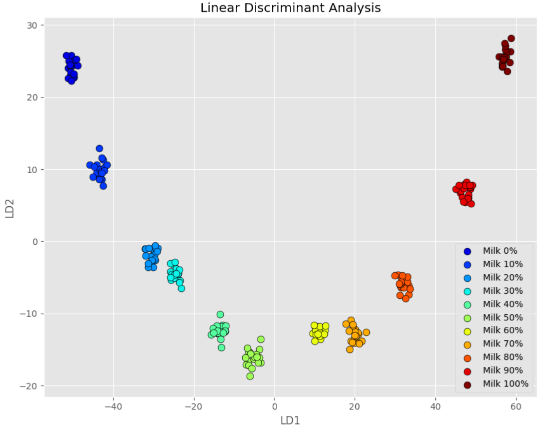
	PCA	**LDA**
Input	correlated	linearly independent
Output	linearly independent	classes
Purpose	dimensional reduction	classification
learning	unsupervised	supervised

similar applications because

- classes and PCs are correlated
- LDA works with weakly correlated data too

Best: PCA *and* LDA

Questions


Linear Discriminant Analysis
LD2
LD1
30
20
10
0
−10
−20
−40
−20
0
20
40
60
Milk 0%
Milk 10%
Milk 20%
Milk 30%
Milk 40%
Milk 50%
Milk 60%
Milk 70%
Milk 80%
Milk 90%
Milk 100%