Principal Component Analysis
Linear Discriminant Analysis

Lecturer:
Jan Čech

Authors:
Ondřej Drbohlav, Jiří Matas

Centre for Machine Perception
Czech Technical University, Prague
http://cmp.felk.cvut.cz

Last update: 21.12.2017

Principal Component Analysis (PCA), Introduction

- Alternative name: Karhunen–Loeve transform
- Used for: data approximation, identifying sources of variance in the data

Maximum variance formulation (1/3)

Let the data be $\{\mathbf{x}_i \in \mathbb{R}^D \mid i = 1, 2, ..., N\}$. Let their mean be $\overline{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i$.

Let us find the unit vector $\mathbf{u} \in \mathbb{R}^D$ to project to such that the variance $J(\mathbf{u})$ of the projected data is *maximized*. The projection $\mathbf{x}'_i$ of an $\mathbf{x}_i$ to one-dimensional linear subspace generated by $\mathbf{u}$ is given by

$$\mathbf{x}'_i = \mathbf{u}\, a_i\,, \qquad a_i = \mathbf{u}^{\mathrm{T}}\mathbf{x}_i\,, \quad (\mathbf{u}^{\mathrm{T}}\mathbf{u} = 1)\,. \tag{1}$$

The variance is an average squared deviation of values from mean, $\frac{1}{N}\sum_{i=1}^{N}(a_i - \overline{a})^2$, $\overline{a} = \frac{1}{N}\sum_{i=1}^{N} a_i$. Substituting $a_i = \mathbf{u}^{\mathrm{T}}\mathbf{x}_i$, the mean is $\overline{a} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{u}^{\mathrm{T}}\mathbf{x}_i = \mathbf{u}^{\mathrm{T}}\overline{\mathbf{x}}$, and the variance $J(\mathbf{u})$ is thus

$$J(\mathbf{u}) = \frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{u}^{\mathrm{T}}\mathbf{x}_i - \mathbf{u}^{\mathrm{T}}\overline{\mathbf{x}}\right)^2 = \frac{1}{N}\sum_{i=1}^{N}\mathbf{u}^{\mathrm{T}}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^{\mathrm{T}}\mathbf{u} = \mathbf{u}^{\mathrm{T}}\mathbf{S}\mathbf{u}\,, \tag{2}$$

where $\mathbf{S}$ is the normalized scatter matrix:

$$\mathbf{S} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^{\mathrm{T}}\,. \tag{3}$$

The Lagrangian of this optimization problem is

$$L(\mathbf{u}, \lambda) = J(\mathbf{u}) + \lambda \underbrace{(1 - \mathbf{u}^{\mathrm{T}}\mathbf{u})}_{\text{constraint}} = \mathbf{u}^{\mathrm{T}}\mathbf{S}\mathbf{u} + \lambda(1 - \mathbf{u}^{\mathrm{T}}\mathbf{u})\,, \tag{4}$$

where λ is the Lagrange multiplier. Taking the derivative w.r.t. the vector $\mathbf{u}$ and setting it to zero gives

$$\frac{\partial L(\mathbf{u}, \lambda)}{\partial \mathbf{u}} = \mathbf{S}\mathbf{u} - \lambda\mathbf{u}_1 = 0\,, \tag{5}$$

and thus

$$\mathbf{S}\mathbf{u} = \lambda\mathbf{u}\,. \tag{6}$$

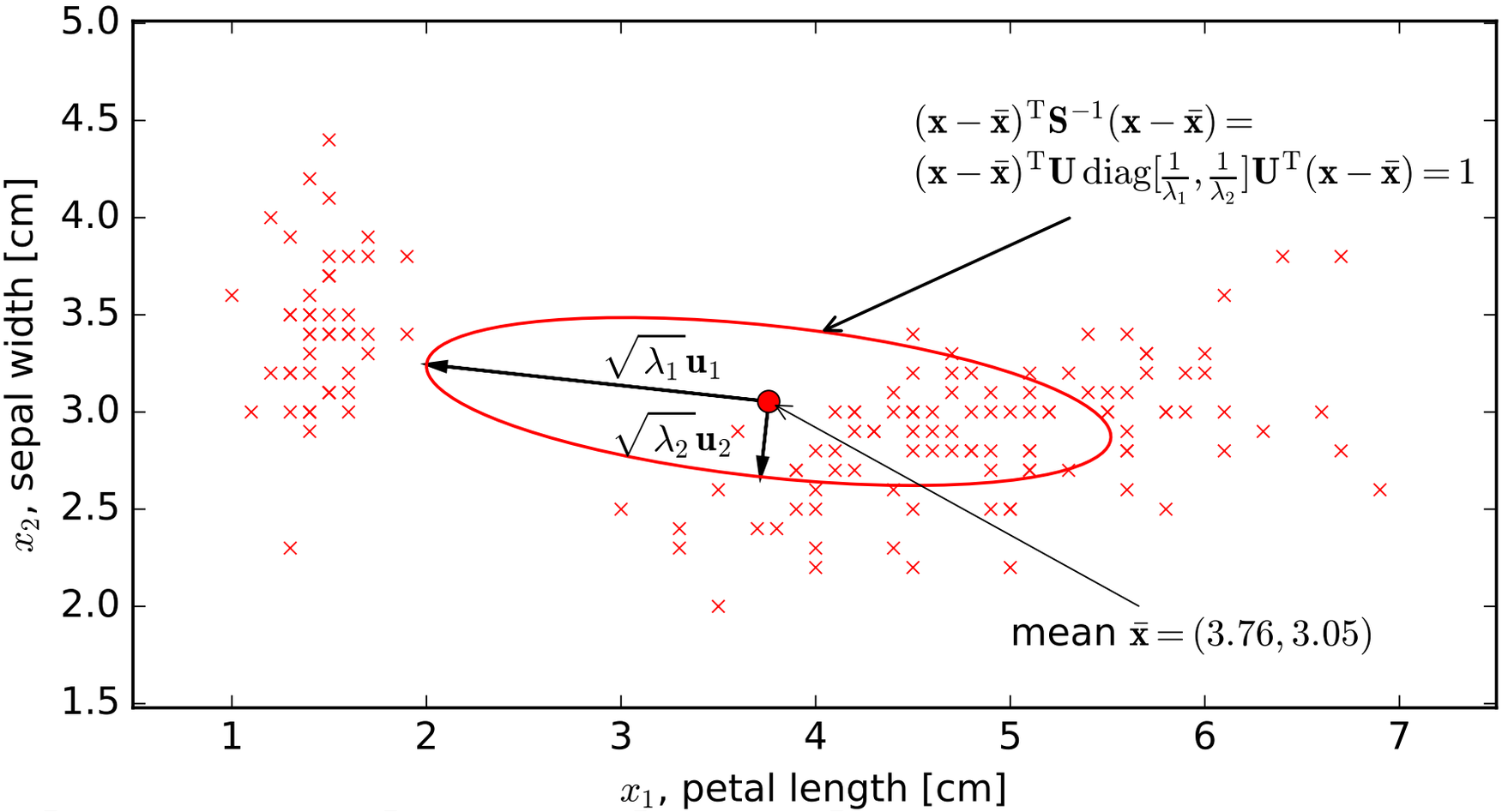
This is the characteristic equation for the covariance matrix $\mathbf{S}$. Any eigenvalue λ and its corresponding eigenvector $\mathbf{u}$ solves this equation, with variance $J(\mathbf{u})$ equal to:

$$J(\mathbf{u}) = \mathbf{u}^{\mathrm{T}}\mathbf{S}\mathbf{u} = \mathbf{u}^{\mathrm{T}}\lambda\mathbf{u} = \lambda\,. \tag{7}$$

The maximum is attained if λ is the largest eigenvalue of the matrix $\mathbf{S}$ and $\mathbf{u}$ is its corresponding eigenvector.

Example 1 - Iris dataset

Iris dataset: feature vectors are 4-dimensional, here dimensions 2 and 3 used (petal length and sepal width). Data shown as crosses $\times$.



$$\mathbf{S} = \begin{bmatrix} 3.09 & -0.32 \\ -0.32 & 0.19 \end{bmatrix} = [\mathbf{u}_1, \mathbf{u}_2] \begin{bmatrix} \lambda_1 & \\ & \lambda_2 \end{bmatrix} [\mathbf{u}_1, \mathbf{u}_2]^{\mathrm{T}}$$

Eigenvectors: $[\mathbf{u}_1, \mathbf{u}_2] = \begin{bmatrix} -0.99 & -0.11 \\ 0.11 & -0.99 \end{bmatrix}$, eigenvalues: $\lambda_1 = 3.13$, $\lambda_2 = 0.15$

Variance is maximized when data are projected to direction $\mathbf{u}_1$.

Maximum variance formulation (3/3)

We have seen that the variance of a 1-D projection is maximized when data are projected to the direction of the eigenvector of $\mathbf{S}$ corresponding to the largest eigenvalue.

It can be shown that the M-dimensional subspace maximizing the variance of the data is the one formed by M eigenvectors of $\mathbf{S}$ corresponding to M largest eigenvalues.

It can also be shown that maximizing variance is equivalent to minimizing projection approximation error.

Recall that the ML estimate of the Multivariate Normal Distribution is defined by sample mean $\overline{\mathbf{x}} \in \mathbb{R}^D$ and covariance matrix $\mathbf{S} \in \mathbb{R}^{D \times D}$. The pdf is then

$$p(\mathbf{x} \mid \overline{\mathbf{x}}, \mathbf{S}) = \frac{1}{\sqrt{|2\pi\mathbf{S}|}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\overline{\mathbf{x}})^{\mathrm{T}}\mathbf{S}^{-1}(\mathbf{x}-\overline{\mathbf{x}})\right\}. \tag{8}$$

Denote the eigenvectors and eigenvalues of $\mathbf{S}$ by $\mathbf{u}_i$ and λ_i, respectively $(i = 1, 2, ..., D)$ and let $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_D$. Let $\mathbf{U}$ stacks the eigenvectors:

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_D] \tag{9}$$

There holds (characteristic equation)

$$\mathbf{SU} = \mathbf{U\Lambda} = \mathbf{U}\begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_D \end{bmatrix}, \tag{10}$$

and

$$\mathbf{S} = \mathbf{U\Lambda U}^{\mathrm{T}}. \tag{11}$$

The pdf can then be equivalently expressed as

$$p(\mathbf{x} \mid \overline{\mathbf{x}}, \mathbf{S}) = p(\mathbf{y}) = \frac{1}{\sqrt{|2\pi\mathbf{\Lambda}|}} \exp\left\{-\frac{1}{2}\mathbf{y}^{\mathrm{T}}\mathbf{\Lambda}^{-1}\mathbf{y}\right\}, \qquad \mathbf{y} = \mathbf{U}^{\mathrm{T}}(\mathbf{x} - \overline{\mathbf{x}}) \tag{12}$$

The exponent in Eq. (12) is

$$-\frac{1}{2}\sum_{i=1}^{D}\frac{y_i^2}{\lambda_i}. \tag{13}$$

Imagine we approximate the data by PCA using first M eigenvectors (because we can store only M or so numbers per data point, or the allowed number of computations is limited.) How should we approximate the exponent? One option would be to truncate the exponent to M factors only. However, then e. g. a point $\mathbf{y} = k\mathbf{u}_{M+1}$, with arbitrarily high k, would produce a zero exponent.

In that case, it is better to store $M+1$ numbers per point: its coordinates in the basis of the first M eigenvectors (that is, y_1, y_2, ..., y_M) and the approximation error $\Delta = y_{M+1}^2 + ... + y_D^2 = \|y\|^2 - y_1^2 - y_2^2 - ... - y_M^2$. The exponent is approximated as

$$-\frac{1}{2}\sum_{i=1}^{M}\frac{y_i^2}{\lambda_i} - \frac{1}{2}\frac{\Delta}{\lambda}, \tag{14}$$

with $\lambda = \lambda_{M+1}$ being a common choice.

Dimensionality of data can be high, and even higher than number of samples.

Consider dimensionality $D = 1\mathrm{M}$ (one million) and number of samples $N = 100$. All analysis still applies, but it would be wasteful to compute eigenvectors for the 1Mx1M matrix, as its rank will anyway be at most N (thus 100). Let us define $\mathbf{X}$ to be a matrix formed by stacking all the data vectors (after having subtracted the mean from them): $\mathbf{X} = [\mathbf{x}_1 - \overline{\mathbf{x}}, \mathbf{x}_2 - \overline{\mathbf{x}}, ..., \mathbf{x}_N - \overline{\mathbf{x}}]$.

Thus,

$$\mathbf{S} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})^{\mathrm{T}} = \frac{1}{N}\mathbf{X}\mathbf{X}^{\mathrm{T}}\,. \tag{15}$$

The characteristic equation is then

$$\frac{1}{N}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{u} = \lambda\mathbf{u}\,. \tag{16}$$

Left-multiplying both sides by $\mathbf{X}^{\mathrm{T}}$ gives

$$\frac{1}{N}\mathbf{X}^{\mathrm{T}}\mathbf{X}\overbrace{(\mathbf{X}^{\mathrm{T}}\mathbf{u})}^{\mathbf{w}} = \lambda\overbrace{(\mathbf{X}^{\mathrm{T}}\mathbf{u})}^{\mathbf{w}}\,. \tag{17}$$

Thus, $\mathbf{X}^{\mathrm{T}}\mathbf{X}$, which is only 100×100, has exactly the same set of eigenvalues:

$$\frac{1}{N}\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{w} = \lambda\mathbf{w}\,. \tag{18}$$

Left-multiplying now by $\mathbf{X}$, we get

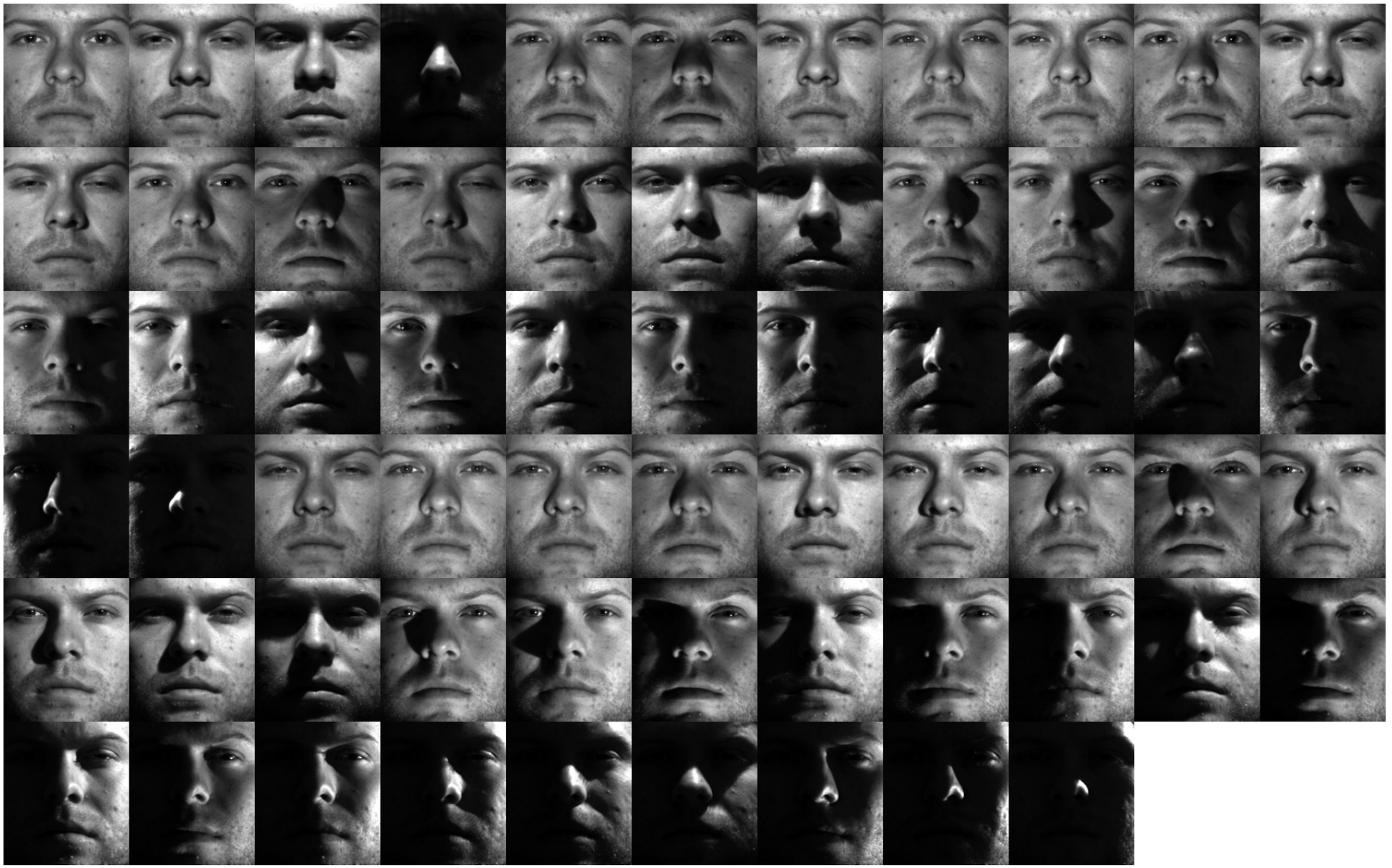
$$\frac{1}{N}\mathbf{X}\mathbf{X}^{\mathrm{T}}(\mathbf{X}\mathbf{w}) = \lambda(\mathbf{X}\mathbf{w})\,. \tag{19}$$

Conclusion: If $D \gg N$, form the matrix $\mathbf{T} = \frac{1}{N}\mathbf{X}^{\mathrm{T}}\mathbf{X}$ and compute its eigenvalues λ's and eigenvectors $\mathbf{w}$. Compute the eigenvectors of $\mathbf{S} = \frac{1}{N}\mathbf{X}\mathbf{X}^{\mathrm{T}}$ as

$$\mathbf{v} = \frac{\mathbf{X}\mathbf{w}}{\|\mathbf{X}\mathbf{w}\|}\,. \tag{20}$$

Example 2 - Yale database (1/5)

images of 38 subjects, each under 64 different illumination conditions:



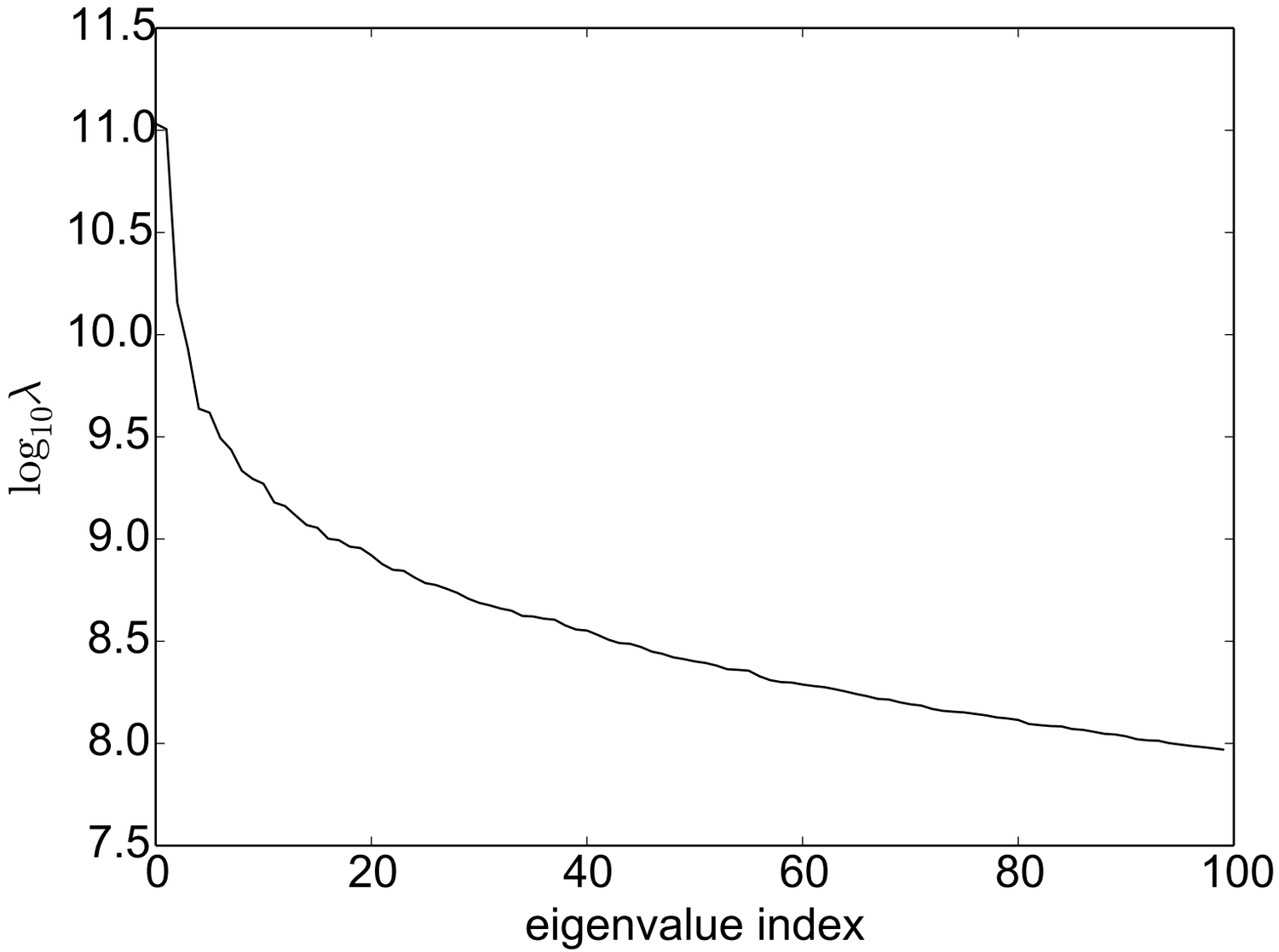
Subject 1, 64 illumination conditions

Example 2 - Yale database (2/5)

images of 38 subjects, each under 64 different illumination conditions:

38 subjects

images of 38 subjects, each under 64 different illumination conditions. Thus, there is $38 \times 64 = 2432$ images in total. Each of them is a feature vector with $192 \times 168 = 32256$ dimensions (pixels). PCA gives the following eigenvalues:



Example 2 - Yale database (4/5)

mean

1st ev

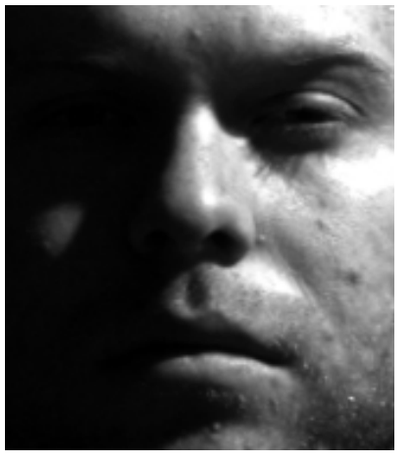
2nd ev

3rd ev

first 72 eigenvectors

Example 2 - Yale database (5/5)

Reconstruction of original vector using eigenvectors



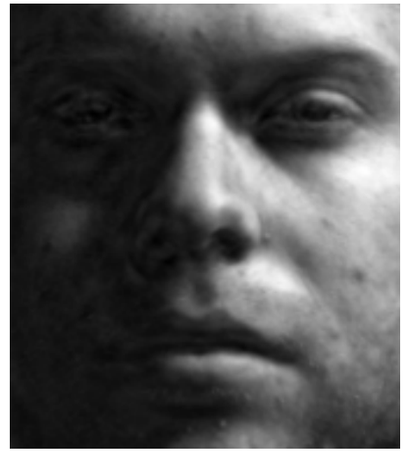
original

mean and 3 evs

mean and 10 evs

mean and 50 evs

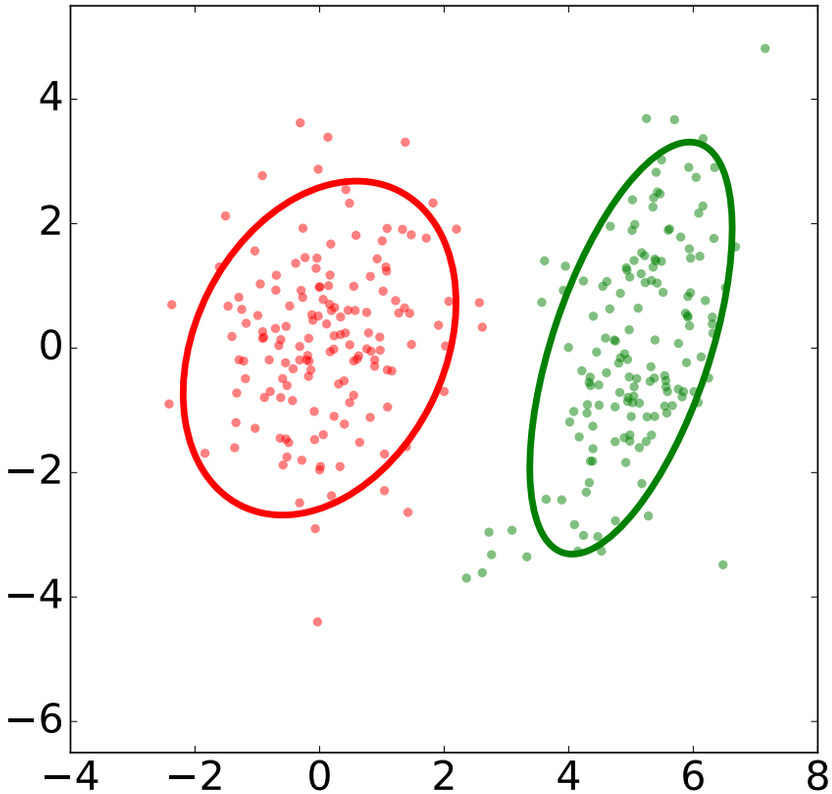
mean and 100 evs



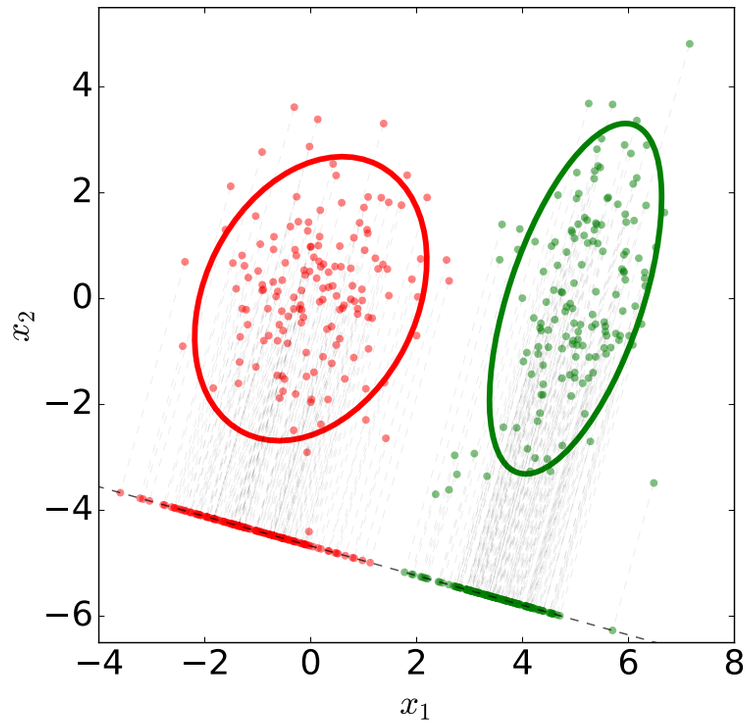
mean and 300 evs

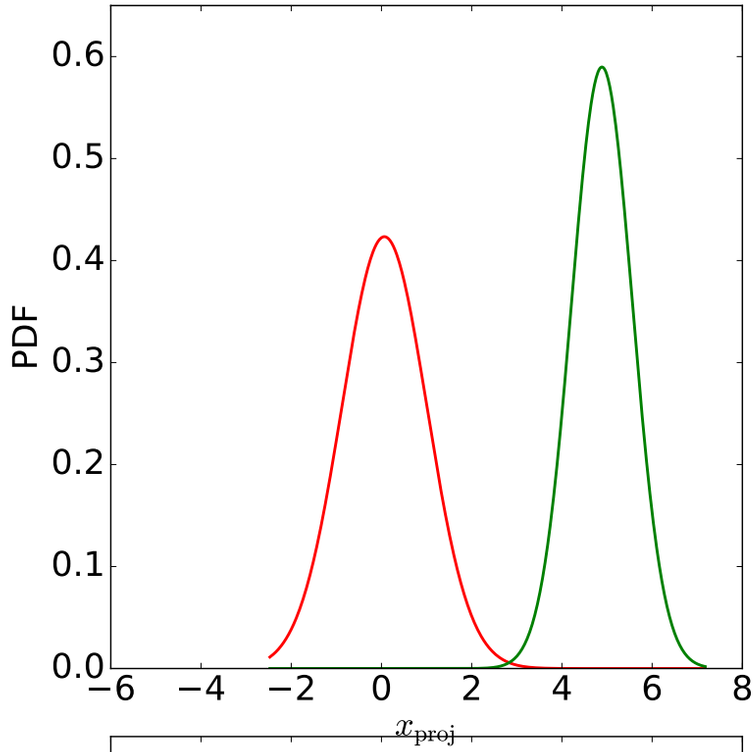
Setting: Classification, training set: N_1 points (class 1) and N_2 points (class 2)
Goal: Project data to a 1D subspace such that a low-error classifier can be constructed.
Approach: Find a direction to project the data to such that the two classes are well separated in this projection.

Example:

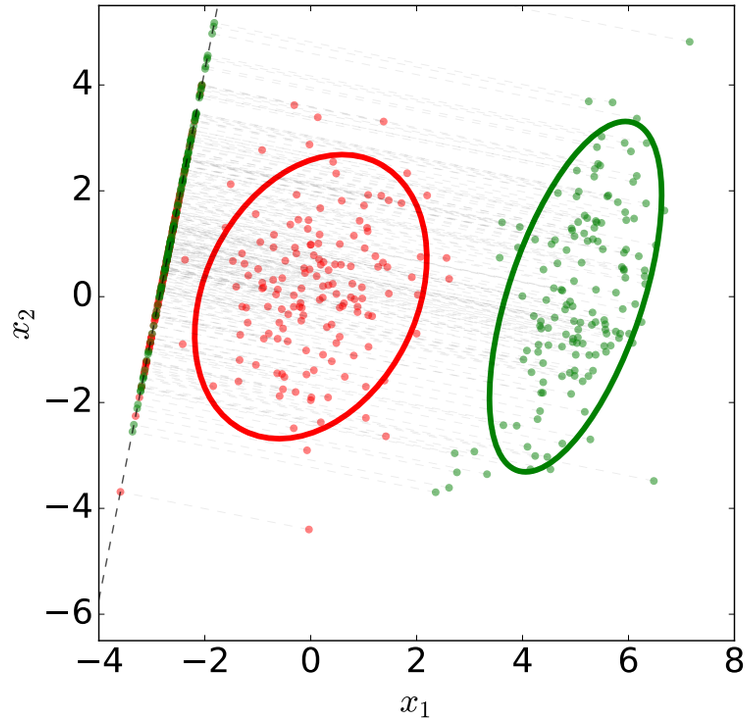


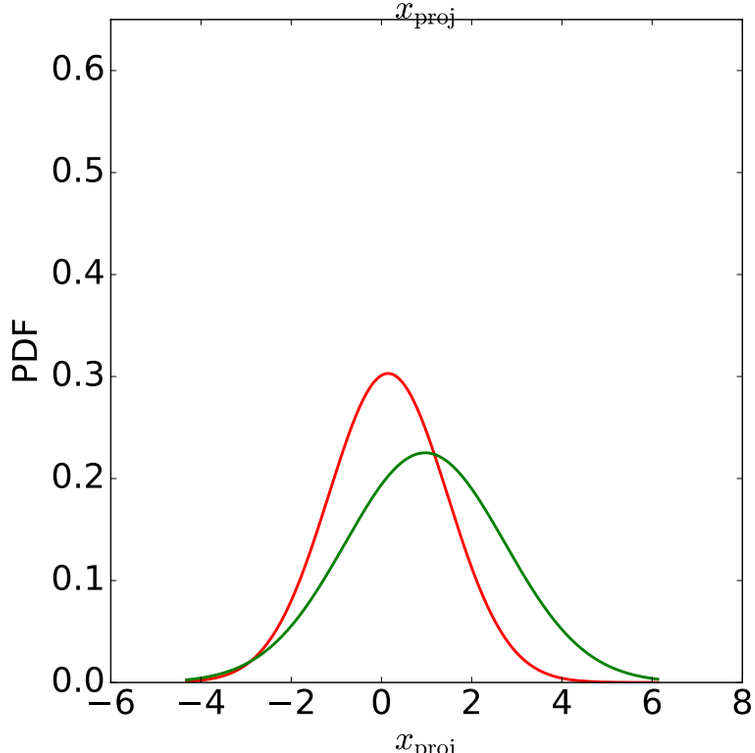
Linear Discriminant Analysis (LDA)





good separation





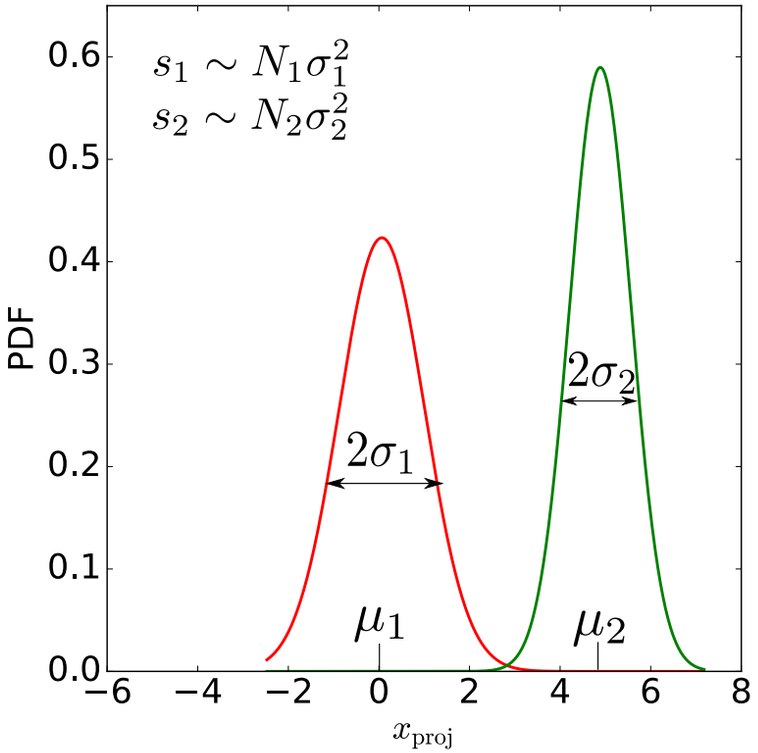
bad separation

Training set: $\mathbf{x}_1^1, ...\mathbf{x}_{N_1}^1$ (class 1), $\mathbf{x}_1^2, ...\mathbf{x}_{N_2}^2$ (class 2).
Separation is higher when:

- the means of projected data are farther apart, and/or
- the scatters of the projected data are smaller.

These two observations combined suggest the following criterion to optimize:

$$\frac{(\mu_1 - \mu_2)^2}{s_1 + s_2} \to \max \tag{21}$$



μ_1, μ_2: mean of projected data

$$\mu_k = \frac{1}{N_k}\sum_{i=1}^{N_k} \mathbf{v}^{\mathrm{T}}\mathbf{x}_i^k = \mathbf{v}^{\mathrm{T}}\overline{\mathbf{x}}_k \qquad (k = 1, 2) \tag{22}$$

s_1, s_2: scatter of projected data

$$s_k = \sum_{i=1}^{N_k} (\mathbf{v}^{\mathrm{T}}\mathbf{x}_i^k - \mathbf{v}^{\mathrm{T}}\overline{\mathbf{x}}_k)^2 \qquad (k = 1, 2) \tag{23}$$

$$\frac{(\mu_1 - \mu_2)^2}{s_1 + s_2} \to \max, \quad \mu_k = \mathbf{v}^{\mathrm{T}}\overline{\mathbf{x}}_k, \quad s_k = \sum_{i=1}^{N_k}(\mathbf{v}^{\mathrm{T}}\mathbf{x}_i^k - \mu_k)^2 \ \ (k = 1, 2) \tag{24}$$

Let us rewrite the criterion in terms of unprojected entities. The nominator:

$$(\mu_1 - \mu_2)^2 = [\mathbf{v}^{\mathrm{T}}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)]^2 = \mathbf{v}^{\mathrm{T}} \underbrace{(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^{\mathrm{T}}}_{\mathbf{S}_b} \mathbf{v} \tag{25}$$

The scatters:

$$s_1 = \sum_{i=1}^{N_1}(\mathbf{v}^{\mathrm{T}}\mathbf{x}_i - \mathbf{v}^{\mathrm{T}}\overline{\mathbf{x}}_1)^2 = \sum_{i=1}^{N_1}\mathbf{v}^{\mathrm{T}}(\mathbf{x}_i - \overline{\mathbf{x}}_1)(\mathbf{x}_i - \overline{\mathbf{x}}_1)^{\mathrm{T}}\mathbf{v} \tag{26}$$

$$= \mathbf{v}^{\mathrm{T}} \underbrace{\left(\sum_{i=1}^{N_1}(\mathbf{x}_i - \overline{\mathbf{x}}_1)(\mathbf{x}_i - \overline{\mathbf{x}}_1)^{\mathrm{T}}\right)}_{\mathbf{S}_1} \mathbf{v} \tag{27}$$

$$s_2 = \mathbf{v}^{\mathrm{T}}\mathbf{S}_2\mathbf{v} \qquad \mathbf{S}_1, \mathbf{S}_2 : \text{scatter matrices for classes 1, 2} \tag{28}$$

$$\frac{(\mu_1 - \mu_2)^2}{s_1 + s_2} \to \text{max}, \quad \mu_k = \mathbf{v}^{\mathrm{T}}\overline{\mathbf{x}}_k, \quad s_k = \sum_{i=1}^{N_k}(\mathbf{v}^{\mathrm{T}}\mathbf{x}_i^k - \mu_k)^2 \ \ (k = 1, 2) \tag{29}$$

Therefore, the criterion can be rewritten as

$$\frac{(\mu_1 - \mu_2)^2}{s_1 + s_2} = \frac{\mathbf{v}^{\mathrm{T}}\mathbf{S}_b\mathbf{v}}{\mathbf{v}^{\mathrm{T}}(\mathbf{S}_1 + \mathbf{S}_2)\mathbf{v}} = \frac{\mathbf{v}^{\mathrm{T}}\mathbf{S}_b\mathbf{v}}{\mathbf{v}^{\mathrm{T}}\mathbf{S}_w\mathbf{v}}, \tag{30}$$

where everything except the to-be-found vector $\mathbf{v}$ is computed from the training data:

$$\mathbf{S}_b \text{ : between-class scatter matrix, } \mathbf{S}_b = (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^{\mathrm{T}} \tag{31}$$

$$\mathbf{S}_w \text{ : within-class scatter matrix, } \mathbf{S}_w = \mathbf{S}_1 + \mathbf{S}_2 \tag{32}$$

$$\mathbf{S}_k = \sum_{i=1}^{N_k}(\mathbf{x}_i^k - \overline{\mathbf{x}}_k)(\mathbf{x}_i^k - \overline{\mathbf{x}}_k)^{\mathrm{T}}, \qquad (k = 1, 2) \tag{33}$$

Let us now solve the maximization task:

$$\mathbf{v}_1 = \underset{\mathbf{v}}{\text{argmax}} \frac{\mathbf{v}^{\mathrm{T}}\mathbf{S}_b\mathbf{v}}{\mathbf{v}^{\mathrm{T}}\mathbf{S}_w\mathbf{v}} \tag{34}$$

Note that there is no need to contrain $\mathbf{v}$ to e.g. unit length, as the scaling in denominator and nominator cancels out.

$$\mathbf{v}_1 = \underset{\mathbf{v}}{\text{argmax}} \frac{\mathbf{v}^{\mathrm{T}}\mathbf{S}_b\mathbf{v}}{\mathbf{v}^{\mathrm{T}}\mathbf{S}_w\mathbf{v}} \tag{35}$$

Note that $\mathbf{S}_b$ is symmetric, positive semi-definite (rank 1) matrix.
Matrix $\mathbf{S}_w$ is symmetric, positive semi-definite.

Assume that $\mathbf{S}_w$ has full rank, thus $\mathbf{S}_w^{-1}$ exists. Let $\mathbf{S}_w^{\frac{1}{2}}$ be the symmetric, positive-definite matrix such that $\mathbf{S}_w = \mathbf{S}_w^{\frac{1}{2}}\mathbf{S}_w^{\frac{1}{2}}$. Let its inverse be denoted $\mathbf{S}_w^{-\frac{1}{2}}$.
Define a substitution

$$\mathbf{z} = \mathbf{S}_w^{\frac{1}{2}}\mathbf{v}\,. \tag{36}$$

Symmetric, positive definite $\mathbf{S}$:
$\mathbf{S} = \mathbf{U}\,\mathrm{diag}[\lambda_1, ..., \lambda_D]\,\mathbf{U}^{\mathrm{T}}$
$\mathbf{U}$: orthogonal, unit columns
$\mathbf{S}^{\frac{1}{2}} = \mathbf{U}\,\mathrm{diag}[\sqrt{\lambda_1}, ..., \sqrt{\lambda_D}]\,\mathbf{U}^{\mathrm{T}}$
$\mathbf{S}^{-\frac{1}{2}} = \mathbf{U}\,\mathrm{diag}[\frac{1}{\sqrt{\lambda_1}}, .., \frac{1}{\sqrt{\lambda_D}}]\,\mathbf{U}^{\mathrm{T}}$
$\mathbf{S}^{-1} = \mathbf{U}\,\mathrm{diag}[\frac{1}{\lambda_1}, ..., \frac{1}{\lambda_D}]\,\mathbf{U}^{\mathrm{T}}$

Using the variable $\mathbf{z}$, the criterion becomes

$$\frac{\mathbf{v}^{\mathrm{T}}\mathbf{S}_b\mathbf{v}}{\mathbf{v}^{\mathrm{T}}\mathbf{S}_w\mathbf{v}} = \frac{\mathbf{z}^{\mathrm{T}}\mathbf{S}_w^{-\frac{1}{2}}\mathbf{S}_b\mathbf{S}_w^{-\frac{1}{2}}\mathbf{z}}{\mathbf{z}^{\mathrm{T}}\mathbf{z}} \tag{37}$$

Let us fix the length of $\mathbf{z}$ to 1 ($\mathbf{z}^{\mathrm{T}}\mathbf{z} = 1$). The denomimator is then a constant, and the criterion is maximized when the nominator is maximized. The latter achieves maximum for the largest eigenvalue λ_1 of matrix $\mathbf{S}_w^{-\frac{1}{2}}\mathbf{S}_b\mathbf{S}_w^{-\frac{1}{2}}$ and the corresponding eigenvector $\mathbf{z}_1$:

$$\mathbf{S}_w^{-\frac{1}{2}}\mathbf{S}_b\mathbf{S}_w^{-\frac{1}{2}}\mathbf{z}_1 = \lambda_1\mathbf{z}_1 \tag{38}$$

(copied from previous slide:)

$$\mathbf{S}_w^{-\frac{1}{2}}\mathbf{S}_b\mathbf{S}_w^{-\frac{1}{2}}\mathbf{z}_1 = \lambda_1\mathbf{z}_1 \tag{39}$$

Taking this $\mathbf{z}_1$, and substituting back, gives the solution $\mathbf{v}_1 = \mathbf{S}_w^{-\frac{1}{2}}\mathbf{z}_1$. Left-multiplying the previous equation by $\mathbf{S}_w^{-\frac{1}{2}}$, we see that

$$\mathbf{S}_w^{-1}\mathbf{S}_b(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{z}_1) = \lambda_1(\mathbf{S}_w^{-\frac{1}{2}}\mathbf{z}_1), \quad \Rightarrow \quad \mathbf{S}_w^{-1}\mathbf{S}_b\mathbf{v}_1 = \lambda_1\mathbf{v}_1\,. \tag{40}$$

Thus $\mathbf{v}_1$ can be computed directly as the eigenvector of $\mathbf{S}_w^{-1}\mathbf{S}_b$ corresponding to the highest eigenvalue, λ_1 (note that $\mathbf{S}_w^{-1}\mathbf{S}_b$ and $\mathbf{S}_w^{-\frac{1}{2}}\mathbf{S}_b\mathbf{S}_w^{-\frac{1}{2}}$ share the eigenvalues).

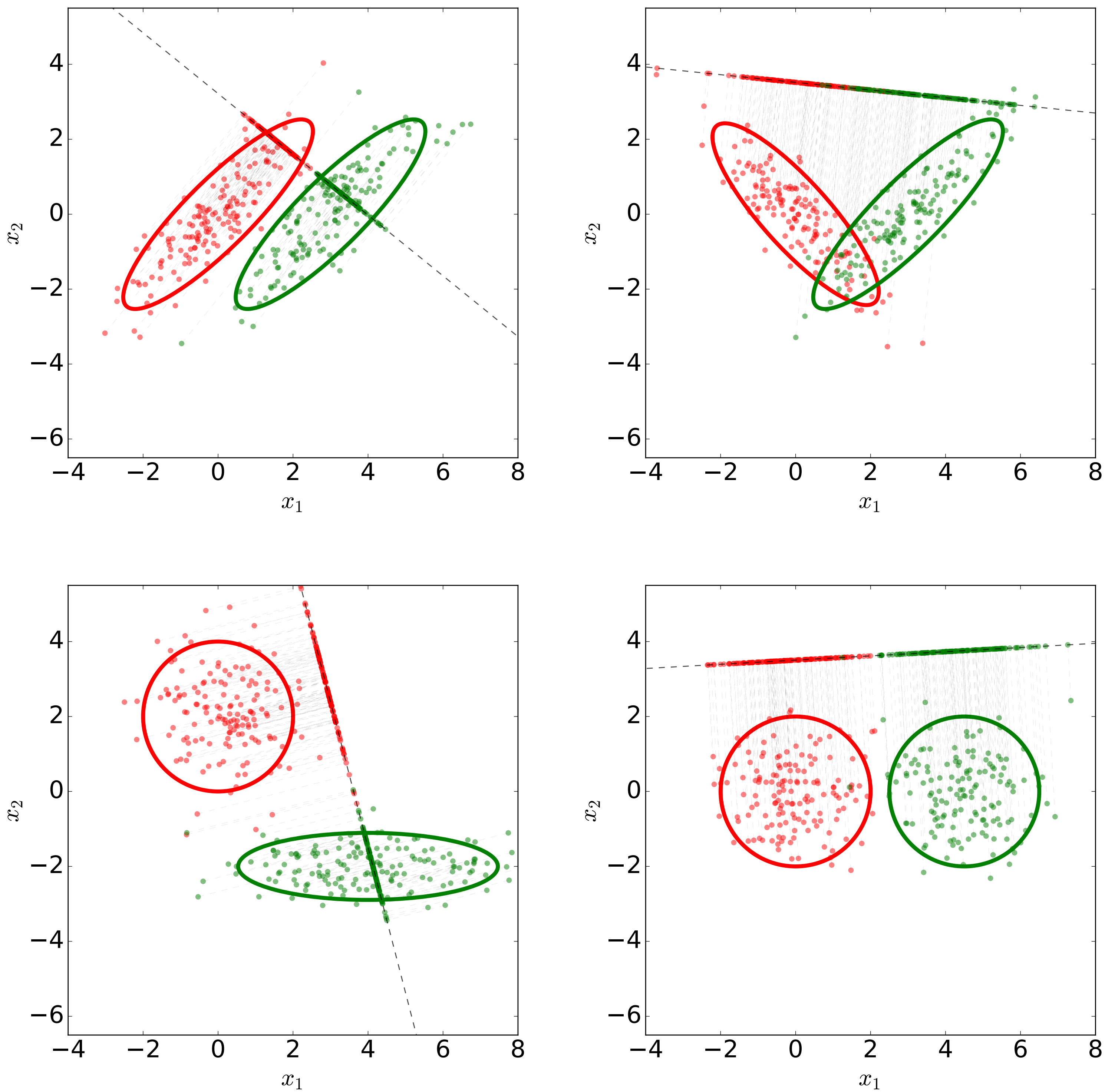
Moreover, $\mathbf{S}_b$ has rank 1. There holds $\mathbf{S}_b = (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^{\mathrm{T}}$, and

$$\mathbf{S}_w^{-1}\mathbf{S}_b\mathbf{v}_1 = \mathbf{S}_w{}^{-1}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)\underbrace{(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^{\mathrm{T}}\mathbf{v}_1}_{\text{a scalar}}, \tag{41}$$

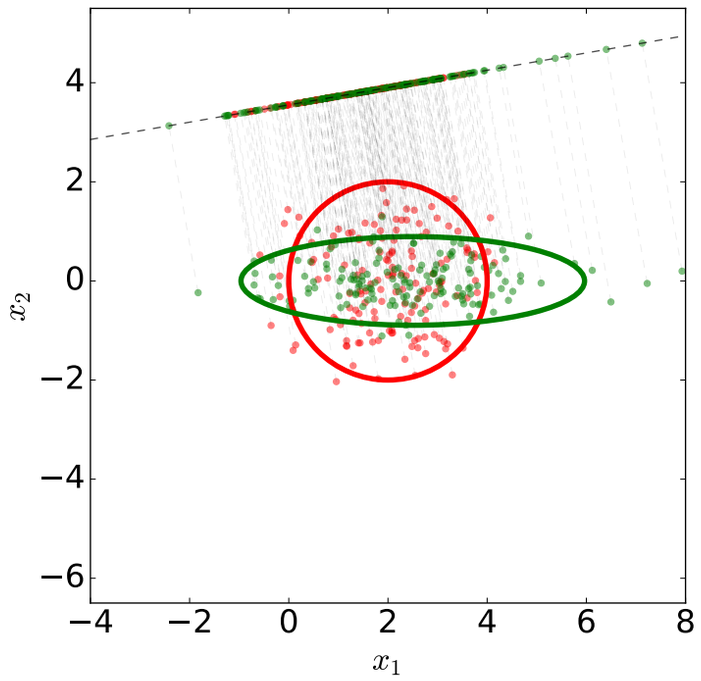
thus the dominant eigenvector (the only one with non-zero eigenvalue) must be

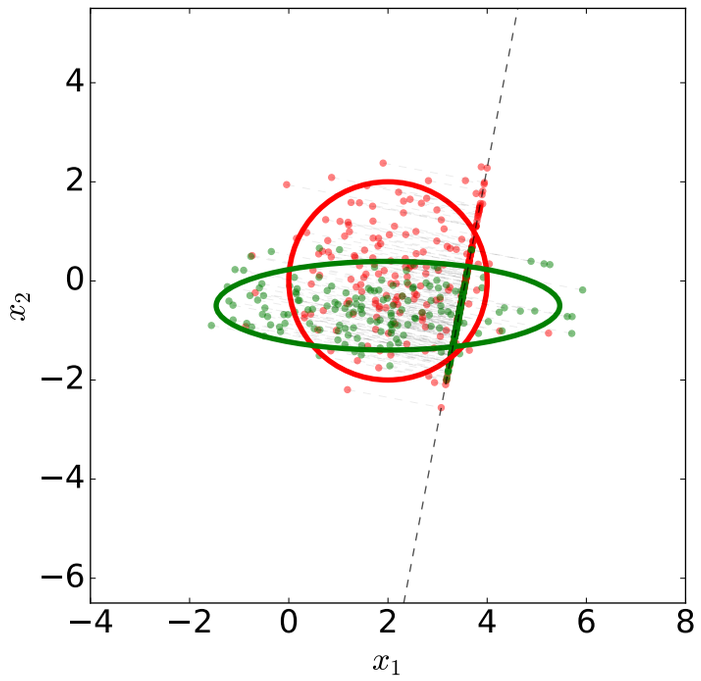
$$\mathbf{v}_1 = \frac{\mathbf{S}_w^{-1}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)}{\|\mathbf{S}_w^{-1}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)\|}\,. \tag{42}$$

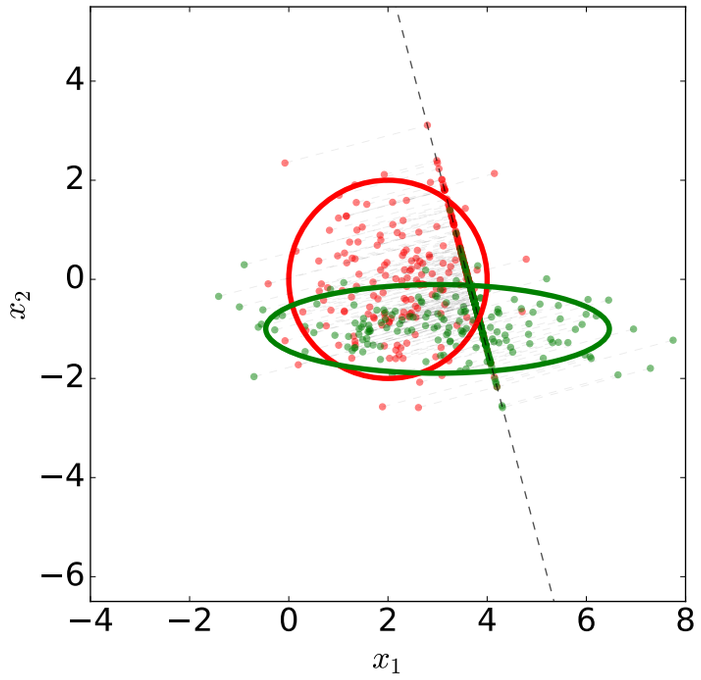
LDA: Examples (1)

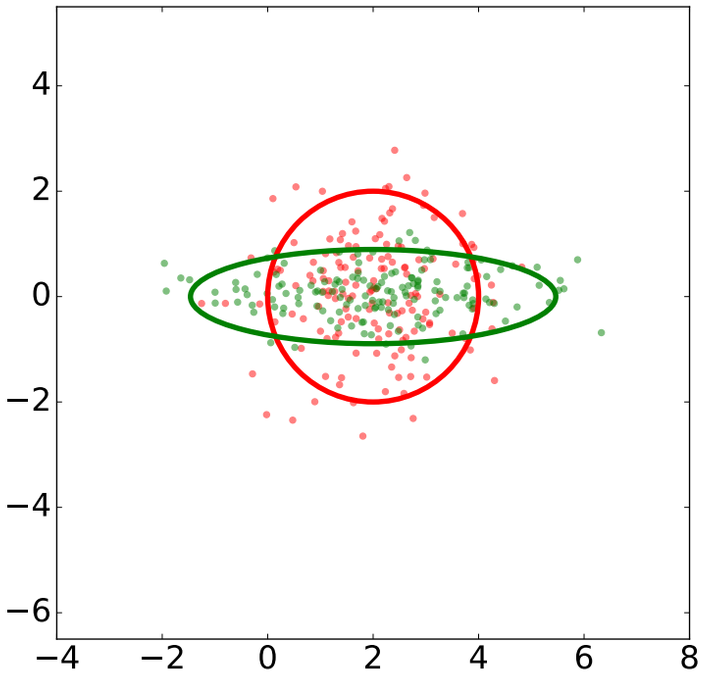


LDA: Examples (2)









Consider the case that the data points $\mathbf{x}$'s are transformed by a non-singular linear transformation $\mathbf{A}$. The entities appearing in formulation and solution of LDA are then transformed as follows:

	points	scatter matrix	inv. scatter m.
original	$\mathbf{x}$	$\mathbf{S}$	$\mathbf{S}^{-1}$
transformed	$\mathbf{Ax}$	$\mathbf{ASA}^{\mathrm{T}}$	$\mathbf{A}^{-\mathrm{T}}\mathbf{S}^{-1}\mathbf{A}^{-1}$

Thus, $\mathbf{v}_1 = \mathbf{S}_w^{-1}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)$ transforms to

$$\mathbf{v}_1' = \mathbf{A}^{-\mathrm{T}}\mathbf{S}_w^{-1}\mathbf{A}^{-1}\mathbf{A}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2) = \mathbf{A}^{-\mathrm{T}}\mathbf{S}_w^{-1}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)\,. \tag{43}$$

The original projected coordinates are

$$\mathbf{v}_1^{\mathrm{T}}\mathbf{x} = (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^{\mathrm{T}}\mathbf{S}_w^{-1}\mathbf{x}\,, \tag{44}$$

and do not change under $\mathbf{A}$, as

$$\mathbf{v}'^{\mathrm{T}}_1\mathbf{x}' = (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^{\mathrm{T}}\mathbf{S}_w^{-1}\mathbf{A}^{-1}\mathbf{Ax} = (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)^{\mathrm{T}}\mathbf{S}_w^{-1}\mathbf{x} = \mathbf{v}_1^{\mathrm{T}}\mathbf{x}\,. \tag{45}$$

Generalization of LDA to multiple classes K

Define:

$$\mathbf{S}_w = \sum_{k=1}^{K} \mathbf{S}_k \quad \text{(sum of class scatters)} \tag{46}$$

$$\mathbf{S}_b = \sum_{k=1}^{K} N_k (\overline{\mathbf{x}}_k - \overline{\mathbf{x}})(\overline{\mathbf{x}}_k - \overline{\mathbf{x}})^{\mathrm{T}} \tag{47}$$

$$\overline{\mathbf{x}}_k = \frac{1}{N_k} \sum_{i=1}^{N_k} \mathbf{x}_i^k \quad \text{(mean of class } k \text{ data)} \tag{48}$$

$$\overline{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i \quad \text{(mean of all data)} \tag{49}$$

Optimization objective: Several options possible, e.g. find a projector $\mathbf{V}$ such that

$$\mathrm{tr}\{(\mathbf{V}\mathbf{S}_w\mathbf{V}^{\mathrm{T}})^{-1}(\mathbf{V}\mathbf{S}_b\mathbf{V}^{\mathrm{T}})\} \to \max \tag{50}$$

Solution: L most significant eigenvectors for the generalized eigenvalue problem:

$$\mathbf{S}_b\mathbf{v} = \lambda\mathbf{S}_w\mathbf{v} \tag{51}$$

Note: $\mathbf{S}_b$ can have rank at most $K-1$, thus at most $K-1$ projection directions will be produced.

Employing MDA:

Useful e.g. when the number of classes K and/or number of data is very high and thus the only information about data which can be used is stored in means and scatters of classes. These are computed in incremental fashion.