flusser@utia.cas.cz

www.utia.cas.cz/people/flusser

Prof. Ing. Jan Flusser, DrSc.

Lecture 5 – Dimensionality Reduction

Problem formulation

- Having D features, we want to reduce their number to n, where $n<<D$

$$(x_1, x_2, \cdots, x_D) \longrightarrow (y_1, y_2, \cdots, y_n)$$

Why to reduce the number of the features?

- **Benefits:** Lower computing complexity
  Improvement of the classification performance

- **Danger:** Possible loss of information

Basic approaches to DR

- **Feature extraction**

  Transform $T: \quad R^D \rightarrow R^n$

  Creation of a new feature space. The features lose their original meaning.

Basic approaches to DR

- **Feature extraction**

  Transform T: $R^D \rightarrow R^n$

  Creation of a new feature space. The features lose their original meaning.

  Example: $n = 1$

$$y_1 = \sum_{i=1}^{D} x_i$$

Basic approaches to DR

- **Feature extraction**

  Transform $T: \quad R^D \rightarrow R^n$

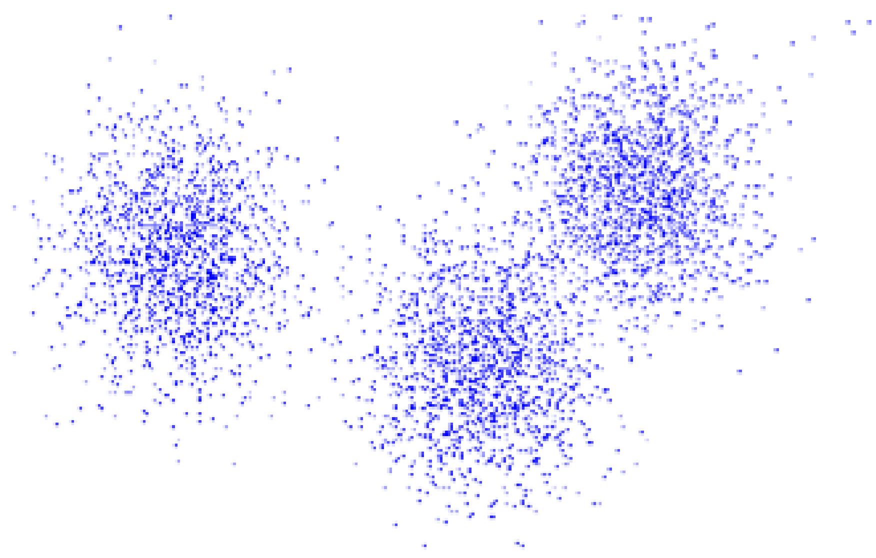
  Creation of a new feature space. The features lose their original meaning.

- **Feature selection**

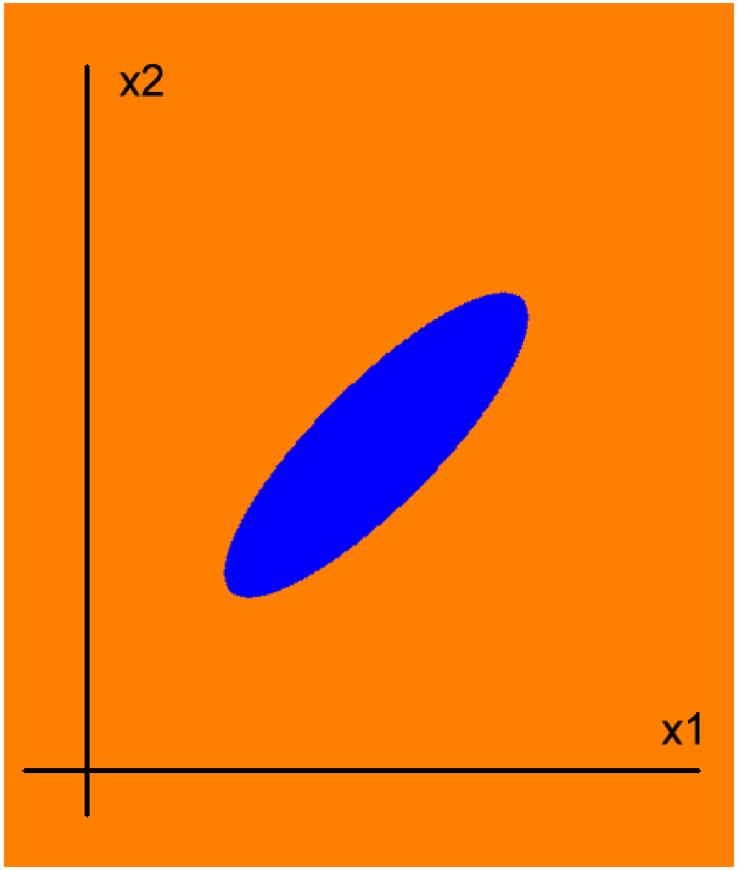
  Selection of a subset of the original features.

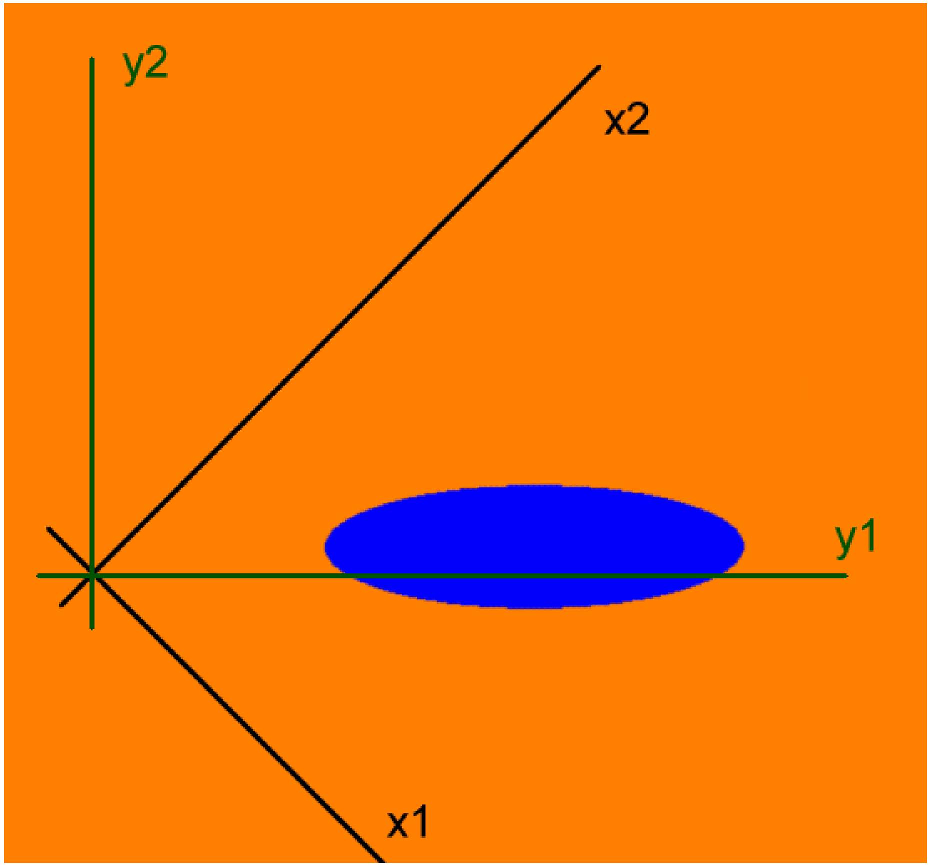
Principal Component Transform
(Karhunen-Loeve Transform)

PCT is a method for "one-class" problem, i.e. for non-structured data (no class representatives, no training sets, just a cloud of data points in the feature space)



Principal Component Transform





- PCT is a rotation of the feature space

$$y = T'x,$$

such that the new features y are **uncorrelated**, i.e. covariance matrix C_y is diagonal.

- This is always possible since the original covariance matrix C_x is symmetric and can be diagonalized in the orthonormal basis of its eigenvectors

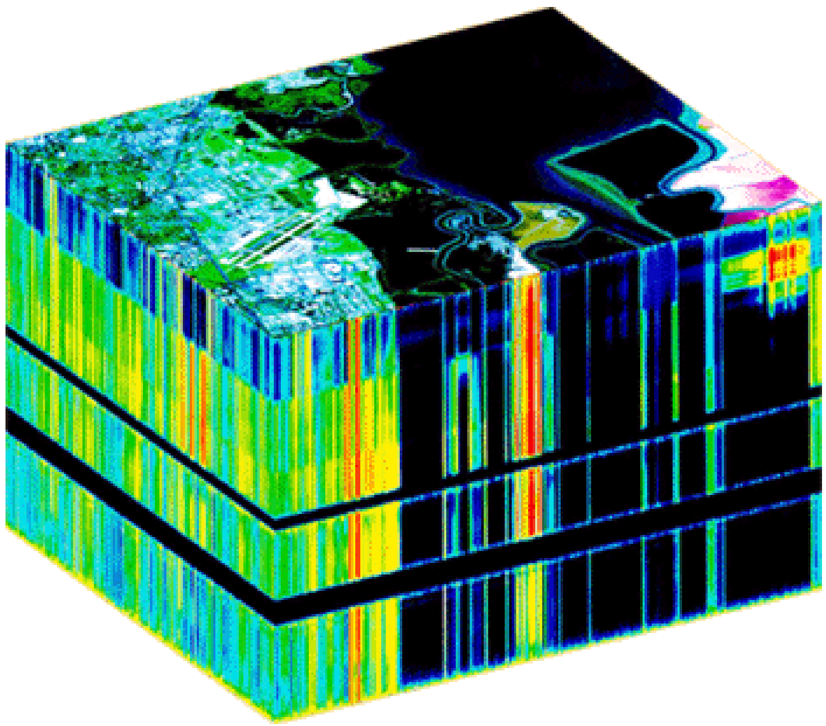
$$C_y = T' C_x T$$

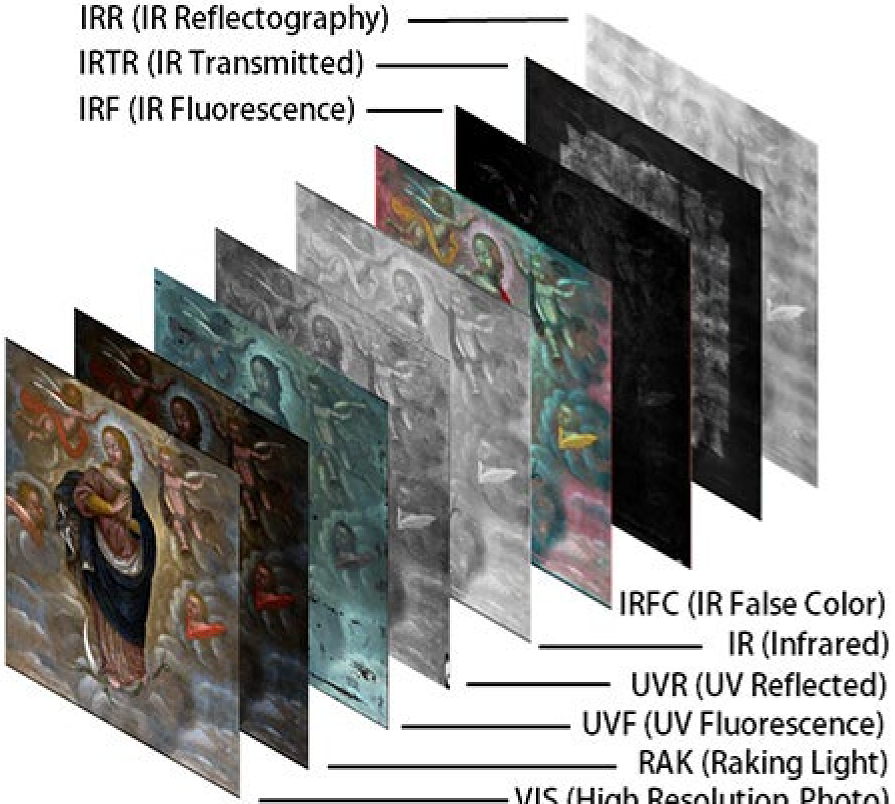
- Features with the highest variances are called **principal components.** First n PC's are kept.

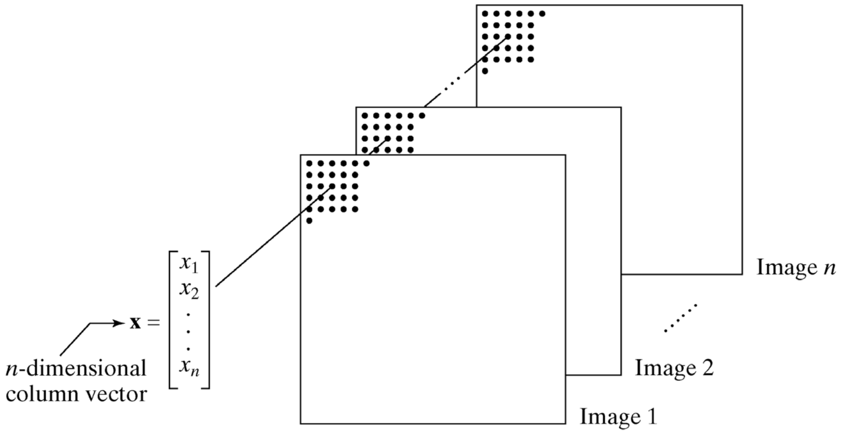
Applications of the PCT

- “Optimal” data representation
- Visualization and compression of multimodal images

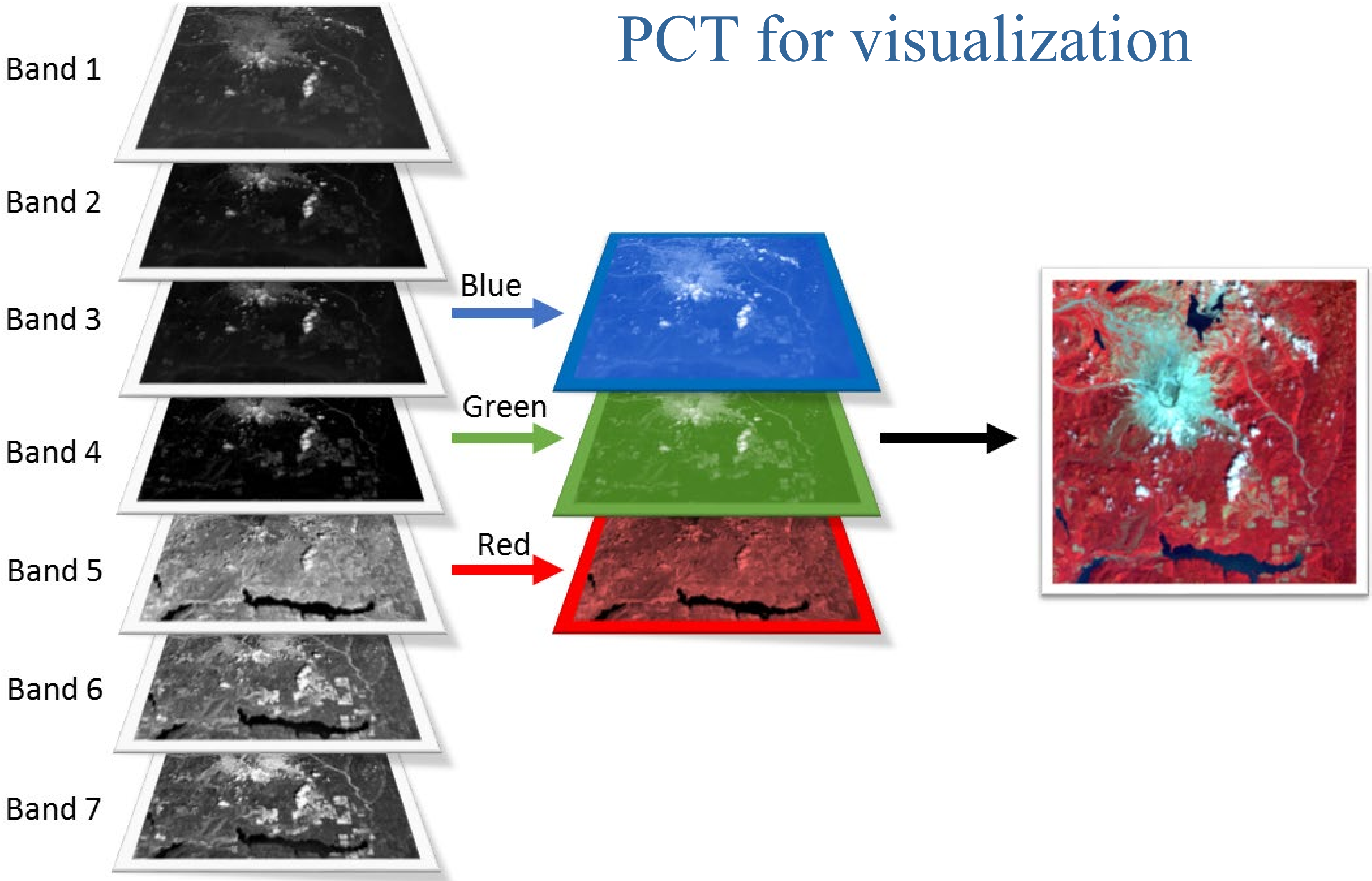
PCT of multispectral images



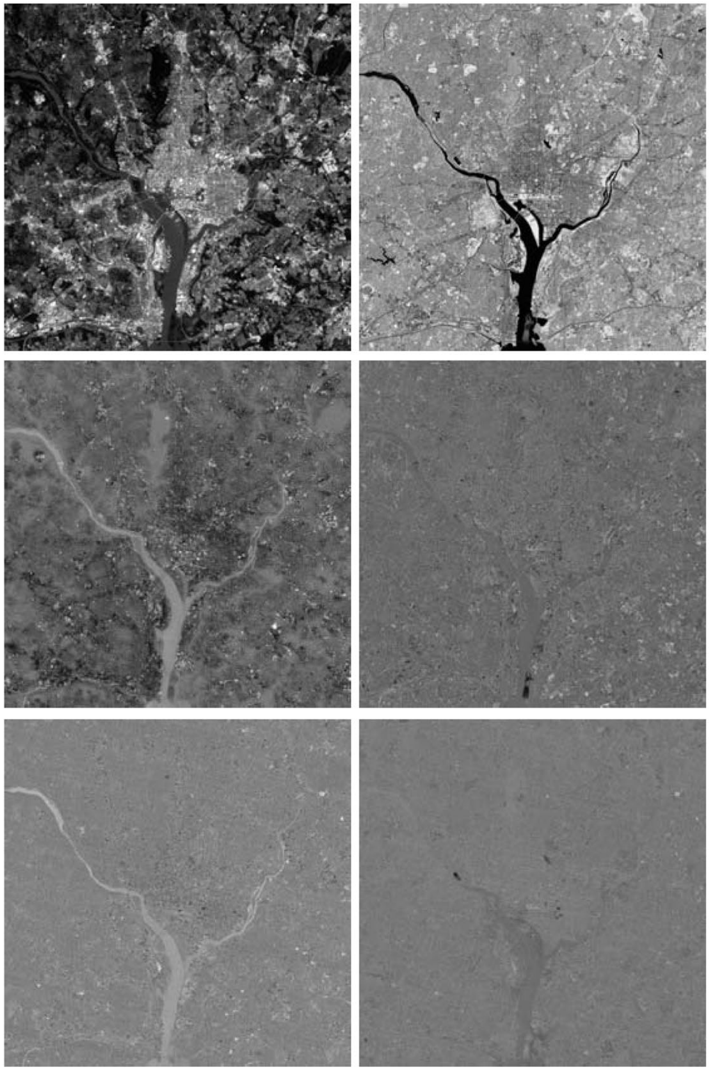




PCT for visualization

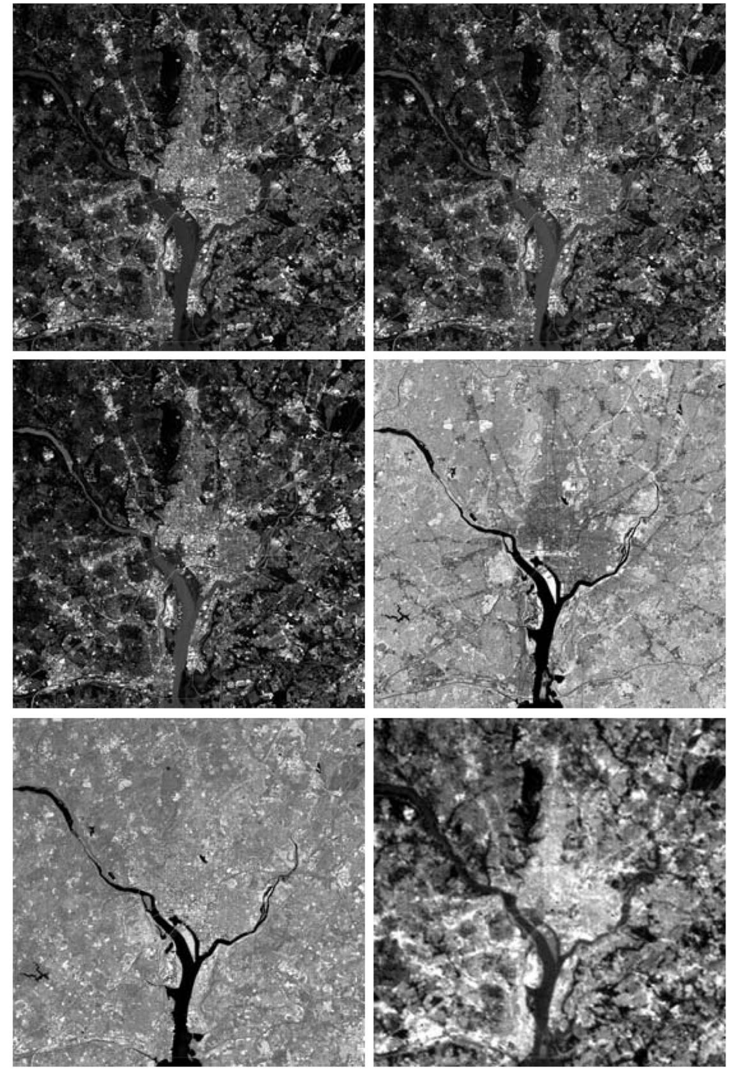


PCT



Reconstruction from two PC's

Original

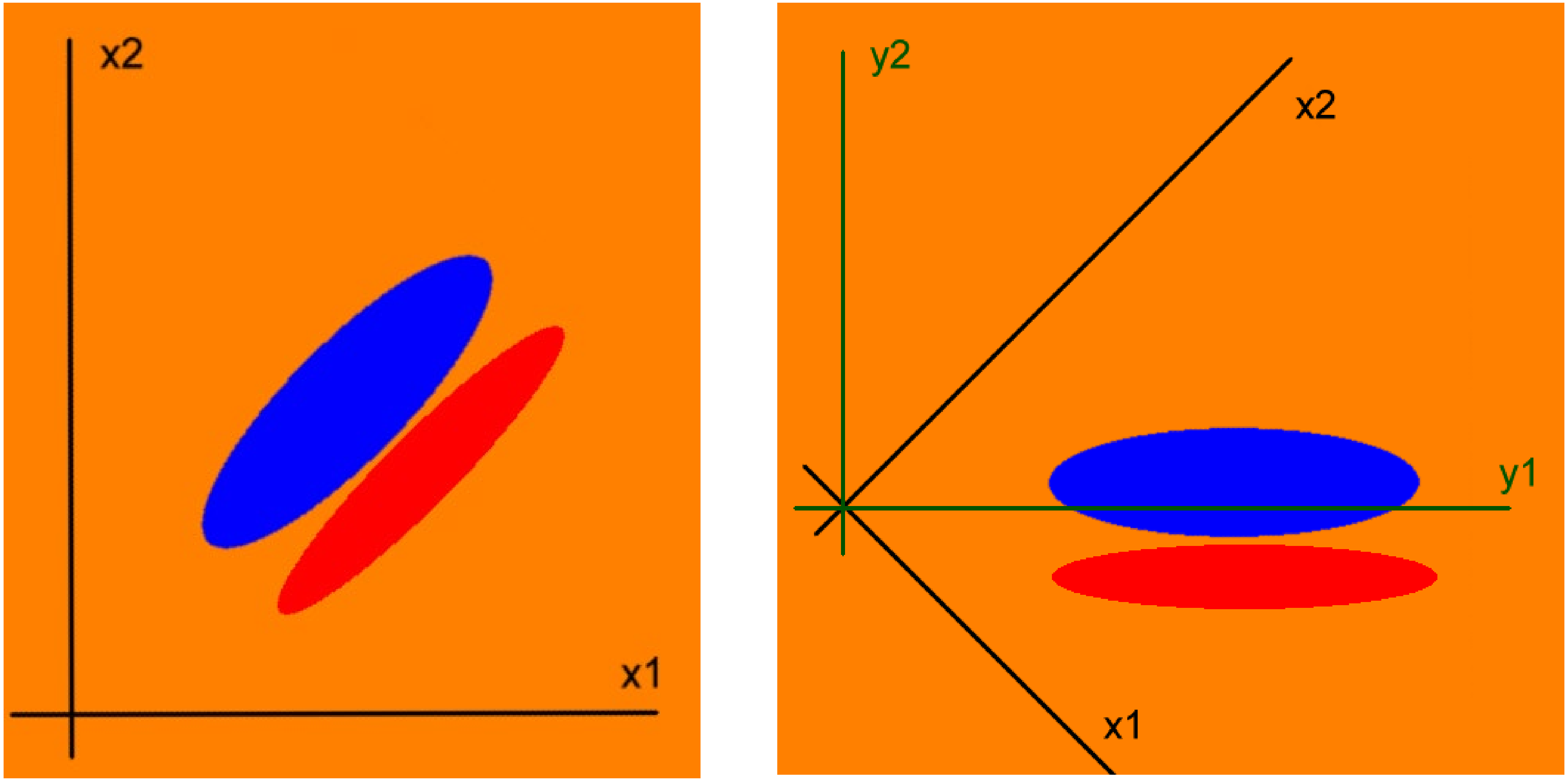


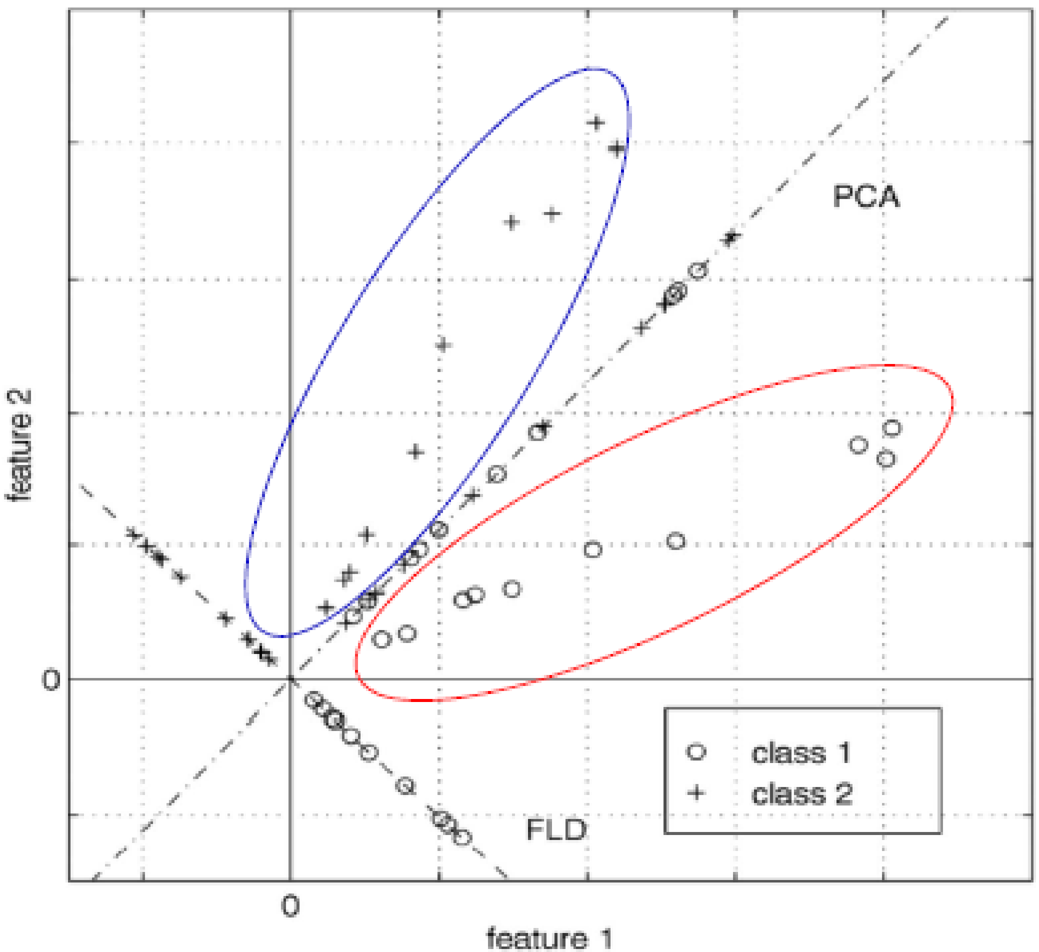
λ_1	λ_2	λ_3	λ_4	λ_5	λ_6
10352	2959	1403	203	94	31

Why is PCT bad for classification purposes?

PCT evaluates the contribution of individual features solely by their variances, which may be different from their **discrimination power.**

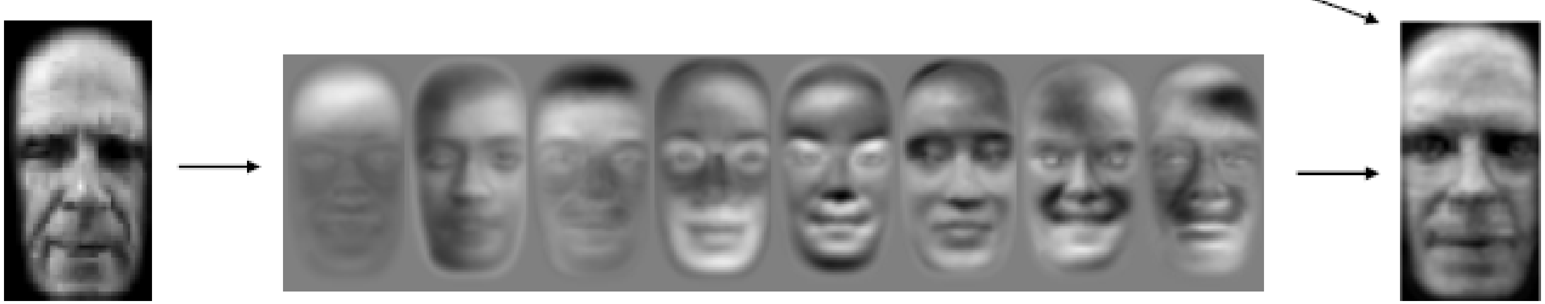
Why is PCT bad for classification purposes?

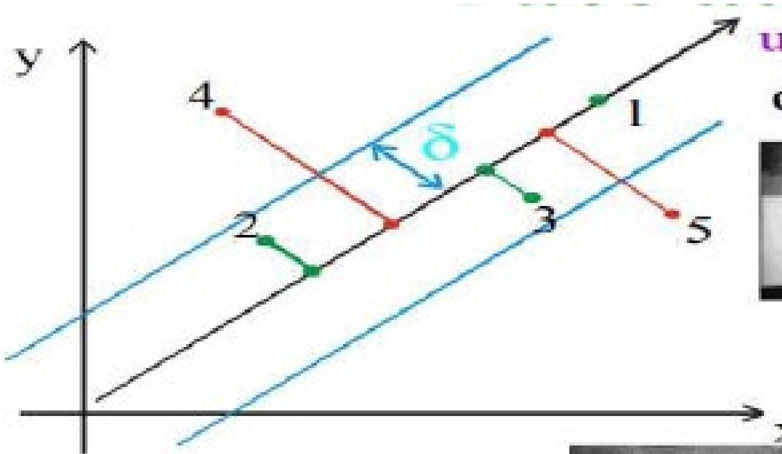


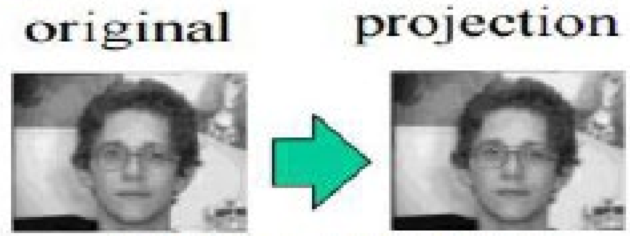

PCA
FLD
feature 1
feature 2
0
0
class 1
class 2

Face recognition by eigenfaces

$$\mathbf{x} \approx \bar{\mathbf{x}} + a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \ldots + a_K\mathbf{v}_\mathbf{K}$$





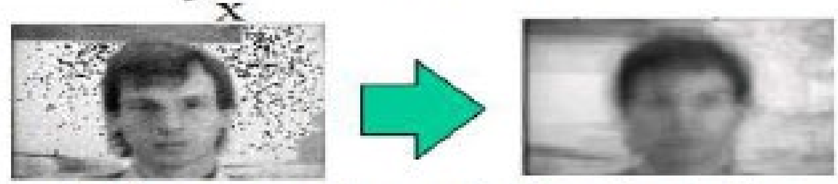


A face, used for training

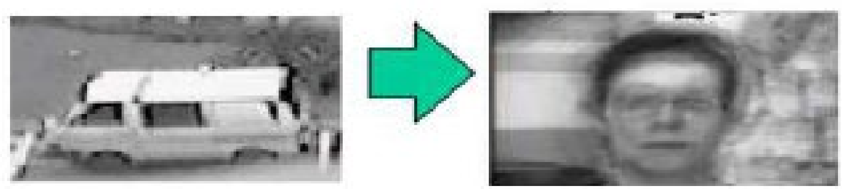
1 *The best case: on the subspace*

2,3 *Close enough*

4,5 *Too far – not a face*



A face, not used for training



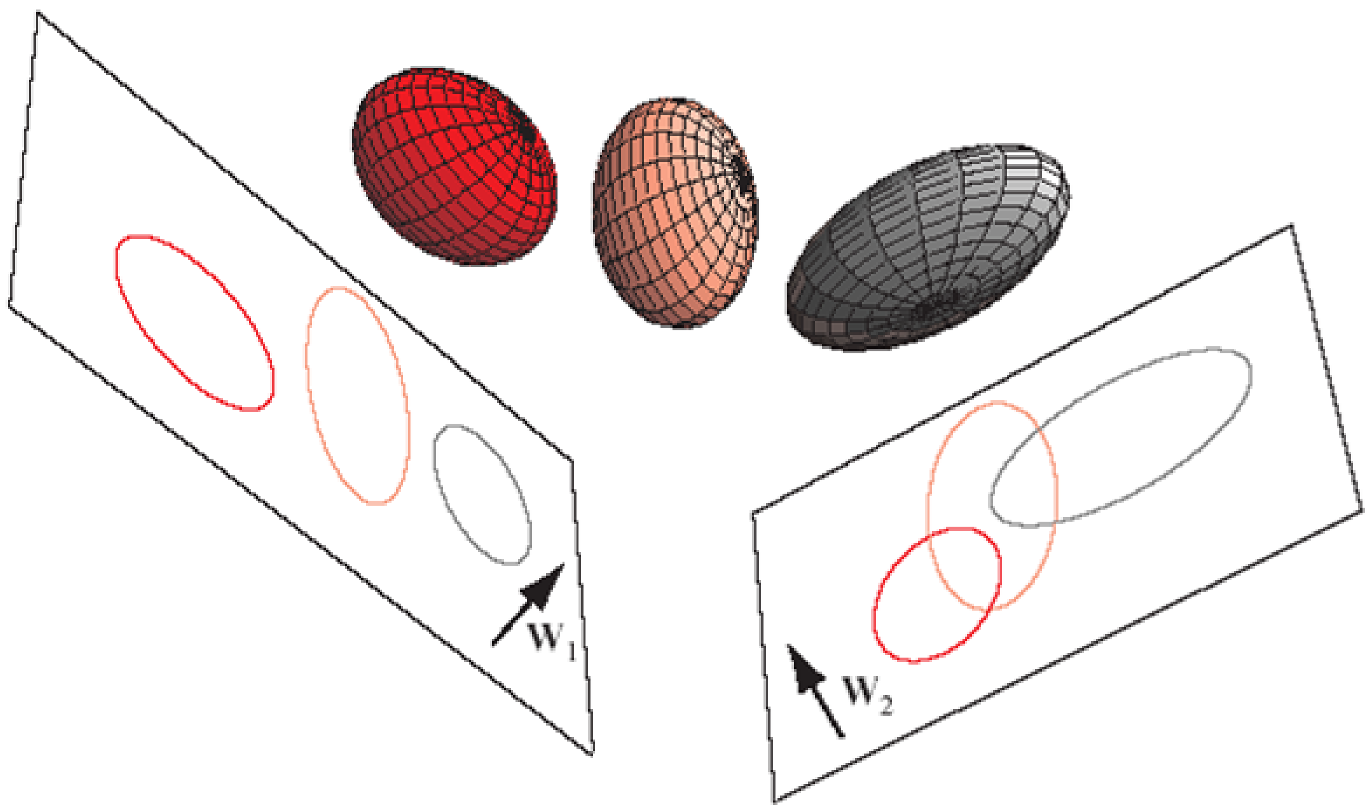
Not a face, not used for training

Multi-class problem

- Training sets for each class are available
- Dimensionality reduction methods for classification purposes must consider the discrimination power (separability) of individual features.
- The goal is to maximize the “distance” between the classes.

Example 1

3 classes, 3D feature space, reduction to 2D



High discriminability   Low discriminability

Feature selection

Two things needed:

- **Discriminability measure** which we want to maximize
- **Selection strategy** (optimization method)

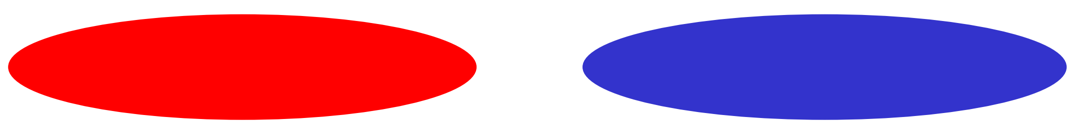
Feature selection → optimization problem

Measures of discriminability between classes

Analogous to those used in clustering but here "clusters" – training sets – are fixed, while the features are subject to selection.

Ward criterion doesn't work.

$$J = \sum_{i=1}^{N} \sum_{\mathbf{x} \in C_i} \|\mathbf{x} - \mathbf{m}_i\|^2$$



Recalling scatter matrices

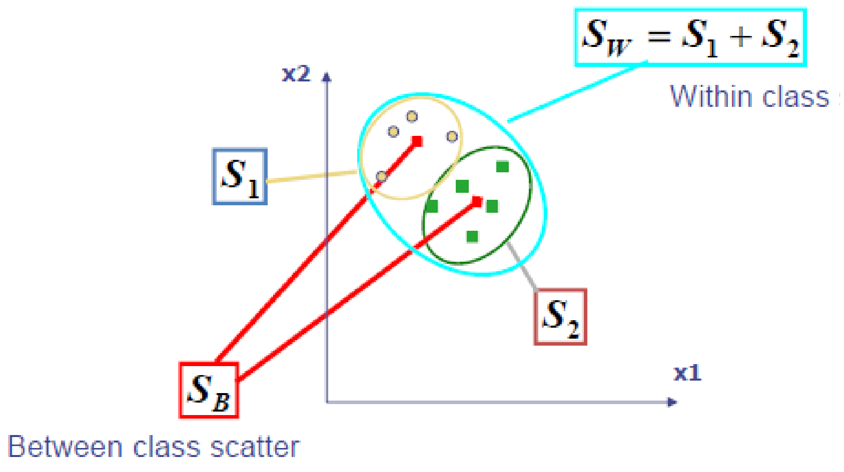
- between cluster matrix

$$B = \sum_{i=1}^{N} n_i(\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^t$$

- within cluster matrix

$$W = \sum_{i=1}^{N} W_i$$

$$W_i = \sum_{\mathbf{x} \in C_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^t$$



The most common discriminability measure

$$\mathrm{tr}(W^{-1}B)$$

If N=2 and both training sets have the same number of elements, then

$$\max \mathrm{tr}(W^{-1}B) \sim \max(\mathbf{m}_1 - \mathbf{m}_2)^t (C_1 + C_2)^{-1} (\mathbf{m}_1 - \mathbf{m}_2)$$

Mahalanobis distance

Bhattacharyya distance

Generalization of the M.D., depends more on the class shapes

$$B = \frac{1}{2}M + \ln \frac{|\frac{1}{2}(C_1 + C_2)|}{\sqrt{|C_1| \cdot |C_2|}}$$

Both *M* and *B* are often used pair-wise also in a multi-class case.

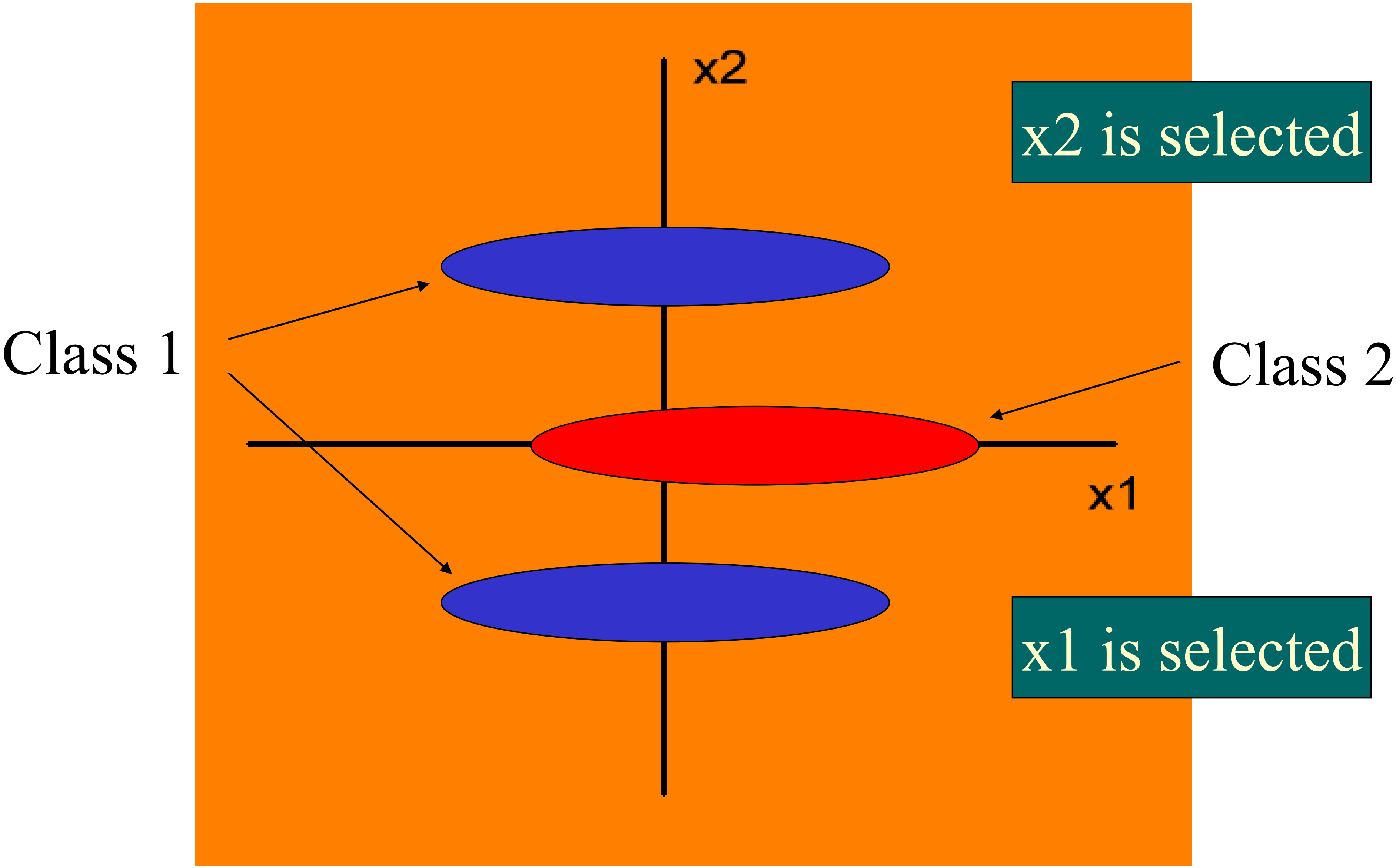
A priori knowledge in feature selection

- The above discriminability measures require **normally distributed** classes (approximation for some **unimodal** classes).

  They are misleading and inapplicable otherwise.

- The normality should be tested (Pearson's test) before.

A two-class example



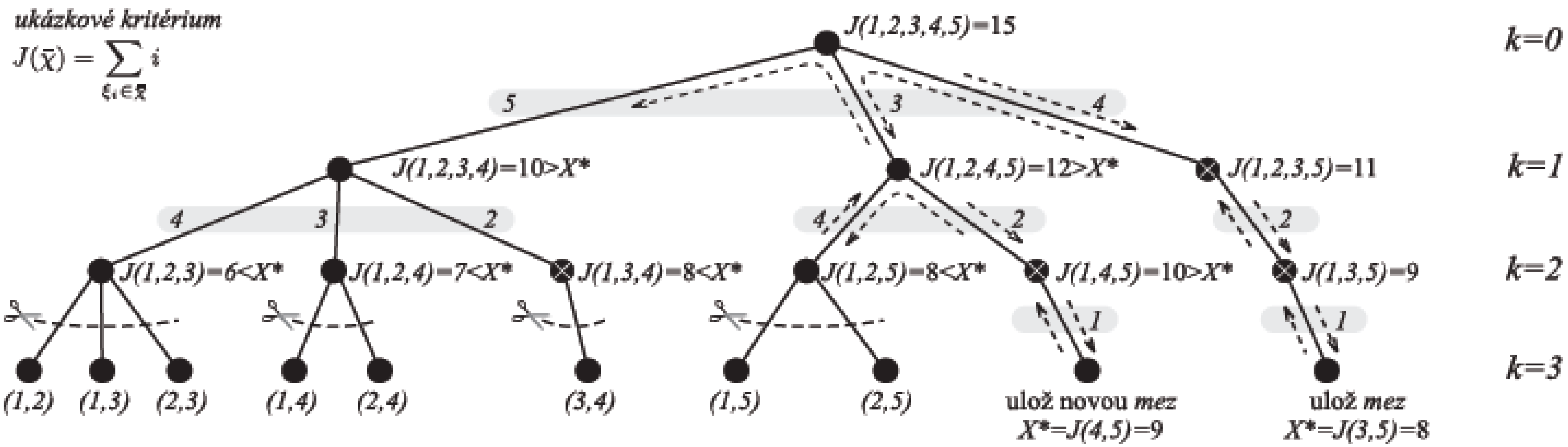
Feature selection algorithms

- **Optimal methods**
  - full search and its modifications
  - complexity $D!/(D-n)!n!$
  - guarantee the global optimum

- **Sub-optimal methods**
  - much faster
  - do not guarantee the global optimum

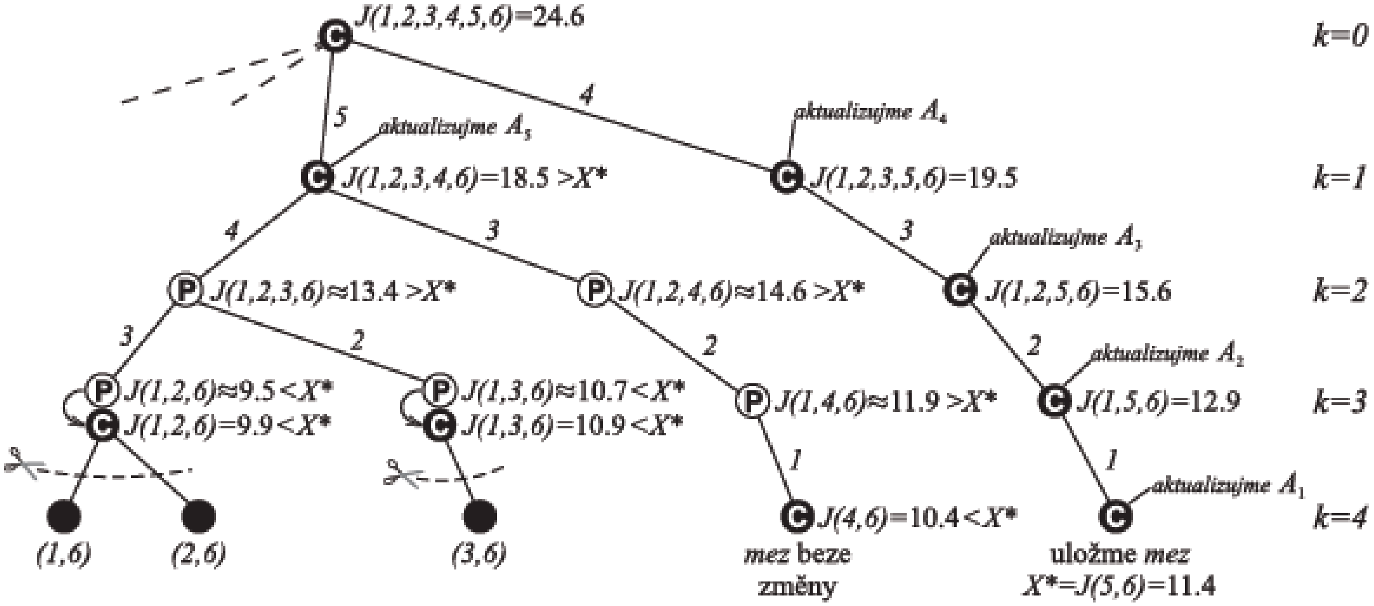
Optimal methods

- Standard full search
- Branch & bound (requires monotonic criteria)



Optimal methods

- Predictive Branch & bound



Demos

Optimal methods

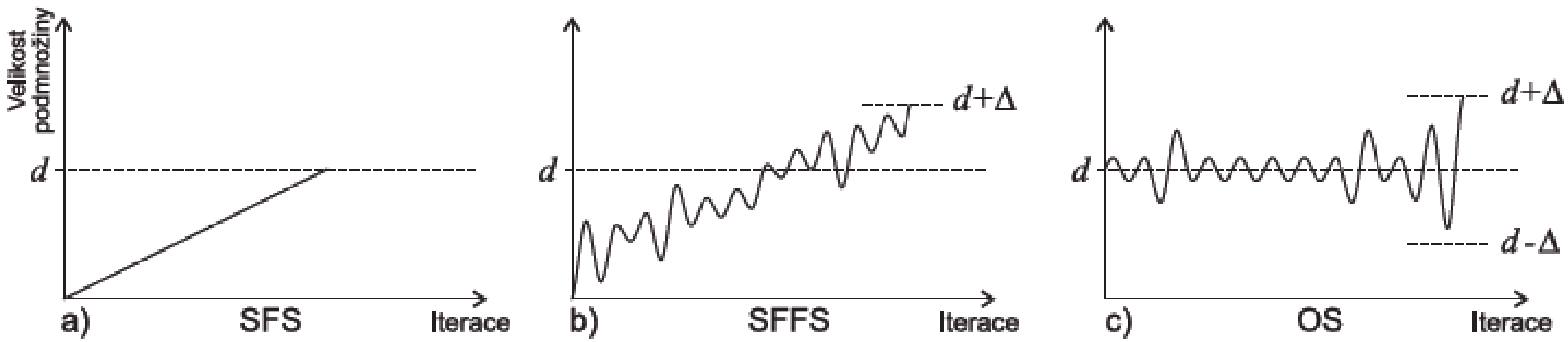
- For comparison of classical and predictive Branch & bound check the 2nd and 3rd demos

http://ro.utia.cas.cz/?q=demo/feature-selection-algorithms

Sub-optimal methods

- Best individual features
  - optimal for uncorrelated data
- Sequential forward/backward selection (nesting effect!)
- "Plus k minus m", $k > m$ (eliminates nesting)
- Floating search
- Oscillating search

Sub-optimal methods



Filters versus Wrappers

- **Filters** optimize the separability of the training set
- **Wrappers** optimize the performance of the particular classifier on the given test set (any selection method can be used, usually much slower)
- Neither filters nor wrappers guarantee optimal performance on independent data but wrappers are believed to be better in this sense

Other remarks

- Stability of the feature selection
- Methods for very high number of dimensions
- Various criteria/selection methods can be fused (similar to combining classifiers)

Thank you !

Any questions ?