CHAPTER 3

Feature Reduction

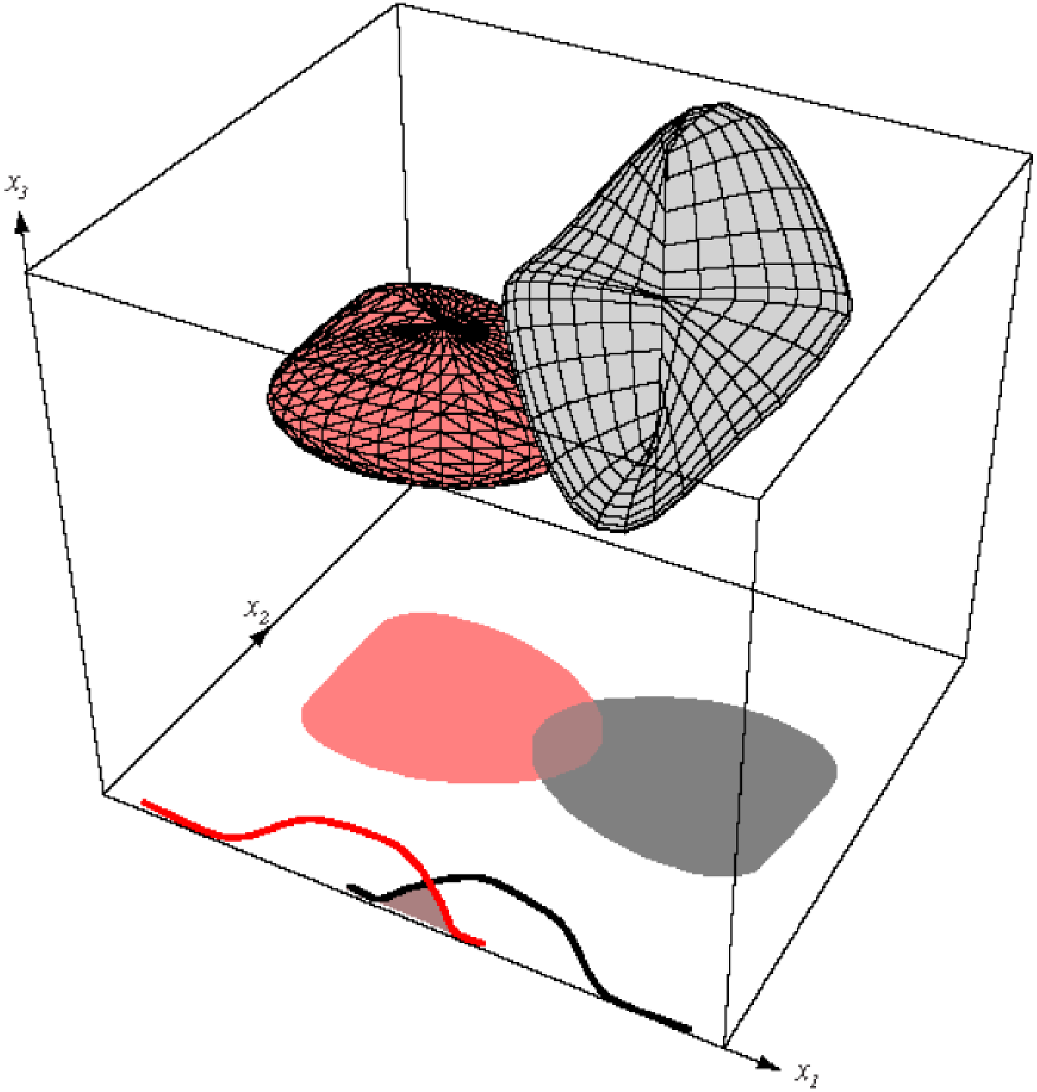
- Introduction
- Hughes Effect
- Statistical Separability Measures
- Sequential Forward/Backward Strategy
- Principal Component Analysis
- Linear Discriminant Analysis

Introduction

- In practical applications, it is not unusual to deal with problems involving tens or hundreds of features.
- Intuitively, it may seem that each feature is useful for at least some of the discriminations.
- In general, if the performance obtained with a given set of features is inadequate, it is natural to consider adding new features.
- Even though increasing the number of features increases the complexity of the classifier, it may be acceptable for an improved performance.

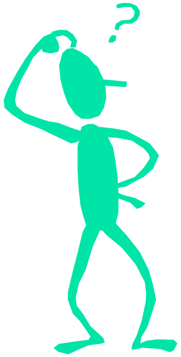
Introduction

There is a non-zero Bayes error in the 1-D x_1 space or the 2-D x_1, x_2 space. However, the Bayes error vanishes in the 3-D x_1, x_2, x_3 space because of non-overlapping densities.



Introduction

- Unfortunately, it has frequently been observed in practice that, beyond a certain point, adding new features leads to worse rather than better performance.
- This is called the curse of dimensionality or Hughes effect.
- There are two issues that we must be careful about:



  - How is the classification accuracy affected by the dimensionality (relative to the amount of training data)?
  - How is the computational complexity of the classifier affected by the dimensionality?

Introduction

- Potential reasons for increase in error include:
  - wrong assumptions in model selection
  - estimation errors due to the finite number of training samples for high-dimensional observations (overfitting)
- Potential solutions include:
  - reducing the dimensionality
  - simplifying the estimation

Introduction

Objectives of feature reduction:

☞ To minimize the implementation cost of the recognition system

☞ To reduce the computational load of the classifier

☞ To overcome the Hughes effect

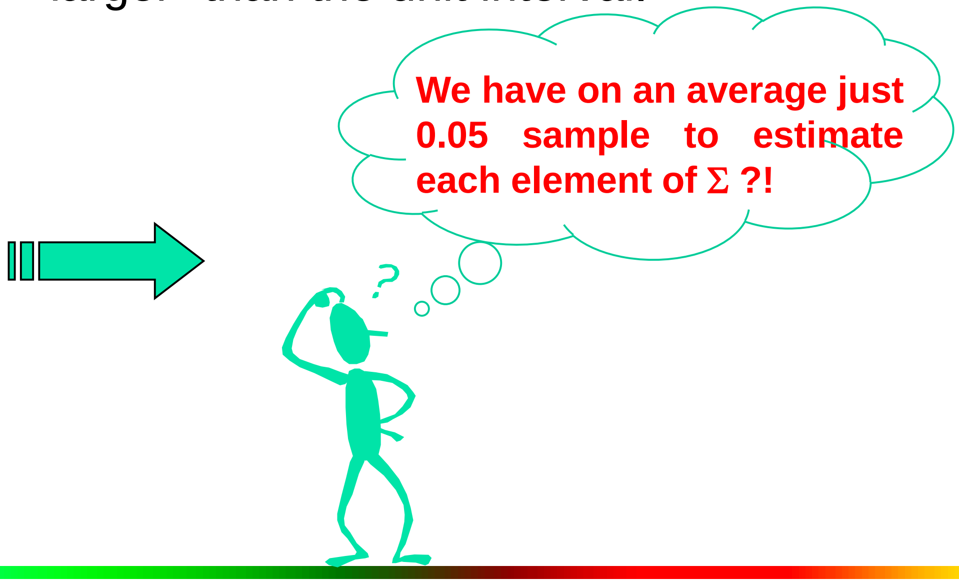
Hughes Effect

- The pdf estimation problem becomes critical when the numbers of training samples and features are unbalanced.
- The balance depends also on the classifier complexity.
- Without such balance, the obtained pdf estimate can result unreliable.
- The Hughes effect is caused by the exponential increase in volume associated with adding extra dimensions to a given space.

Hughes Effect

- Example 1:
  - 100 evenly-spaced sample points suffice to sample a unit interval with no more than 0.01 distance between points.
  - An equivalent sampling of a 10-dimensional unit hypercube with a lattice with a spacing of 0.01 between adjacent points would require 10^{20} sample points!!!
  - Thus, in some sense, the 10-dimensional hypercube can be said to be a factor of 10^{18} "larger" than the unit interval.
- Example 2:

$$\begin{cases} N=1000 \\ n=200 \\ p(\mathbf{x}) \sim N(\mu,\Sigma) \end{cases}$$



We have on an average just 0.05 sample to estimate each element of Σ ?!

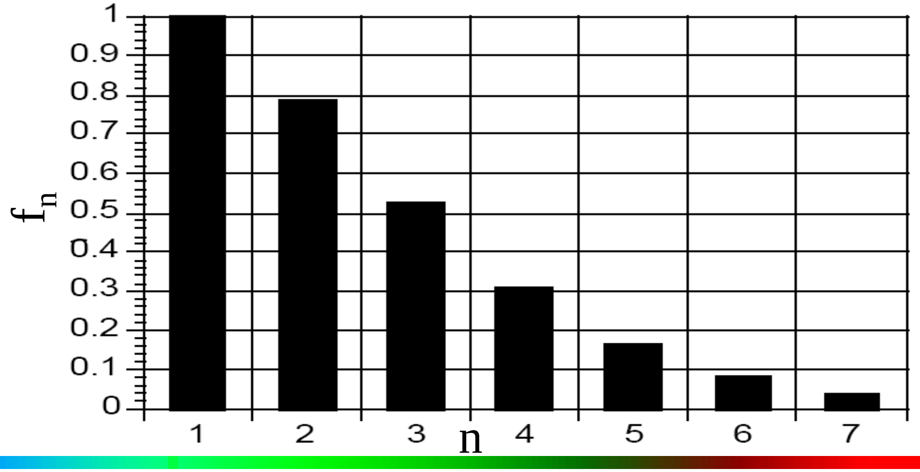
Hughes Effect

- Geometrical Analysis:
  - The volume of a hypersphere of radius r in *n* dimensions is given by: $V_s(r) = \frac{2r^n}{n} \frac{\pi^{n/2}}{\Gamma(n/2)}$

    Gamma function

  - The volume of a hypercube in $[-r, r]^n$ is $V_c(r) = (2r)^n$.
  - The fraction of the volume of a hypersphere inscribed in a hypercube of the same dimension then is:

$$f_n = \frac{V_s(r)}{V_c(r)} = \frac{1}{n2^{n-1}} \frac{\pi^{n/2}}{\Gamma(n/2)} \quad \Rightarrow \quad \lim_{n\to\infty} f_n = 0$$



The volume of the hypercube is increasingly concentrated in the corners as n increases.

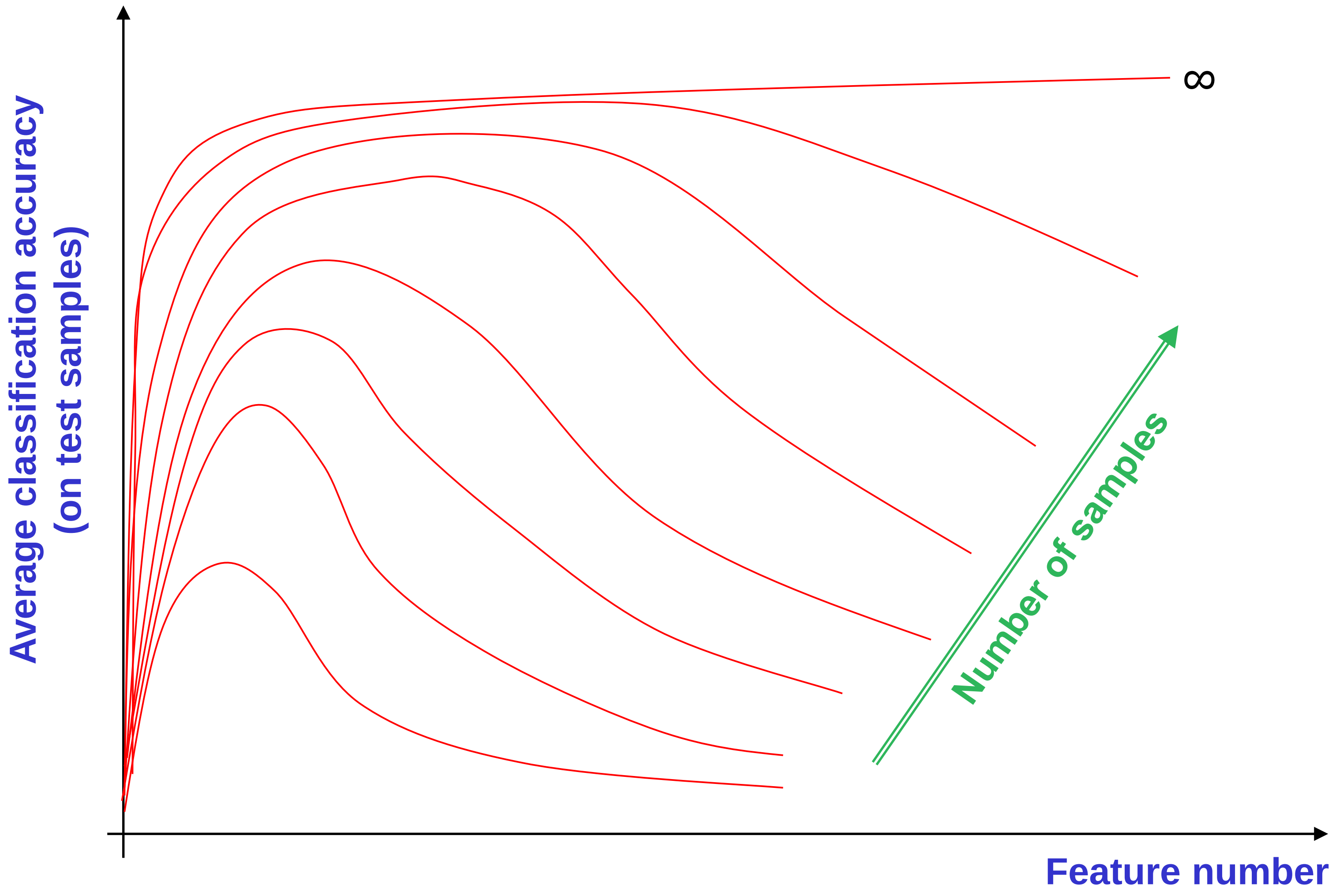
Hughes Effect

- Geometrical Analysis:

☞ High dimensional spaces are mostly empty, which implies that multivariate data in $\Re^n$ is usually in a lower dimensional structure.

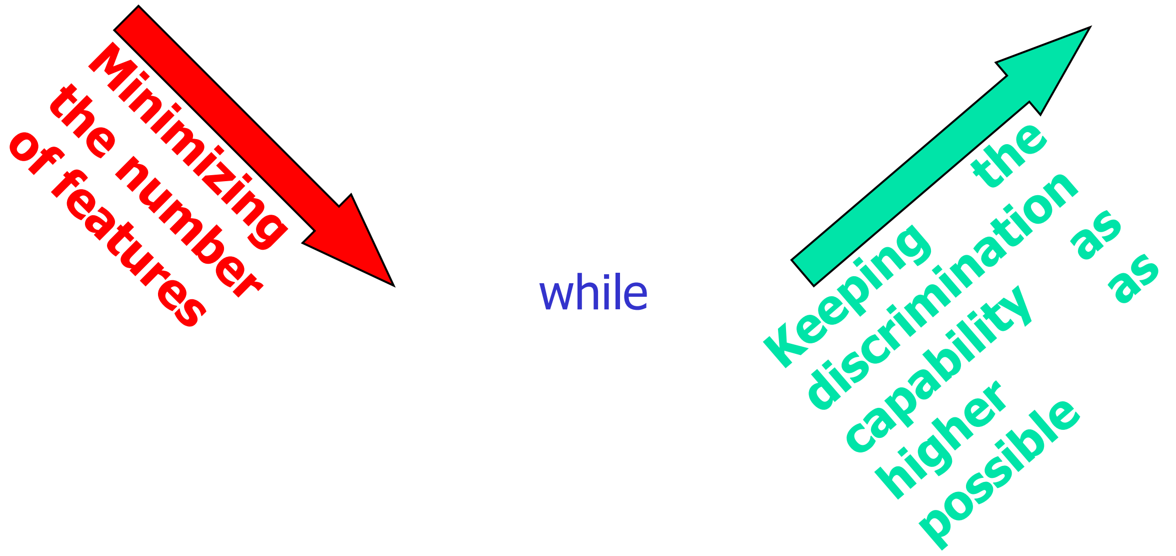
☞ Normally distributed data will have a tendency to concentrate in the tails; similarly, uniformly distributed data will be more likely to be collected in the corners, making density estimation more difficult.

Hughes Effect



Feature Reduction Approaches

- Feature reduction aims at:



Minimizing the number of features

while

Keeping the discrimination capability as higher as possible

- There are two main approaches for feature reduction:
  - Feature reduction by selection (*feature selection*)
  - Feature reduction by transformation (*feature extraction*)

Feature Selection

Problem Formulation

- Let $\boldsymbol{F} = \{\mathrm{x}_1, \ldots, \mathrm{x}_n\}$ be the set of *n* available features.
- In the following, we will denote by f_k and $f_{\underline{k}}$ the *k*-th selected and discarded features from $\boldsymbol{F}$, respectively.
- The goal is to select a subset $\boldsymbol{F^*}$ composed of *m*<*n* features such that:

$$\boldsymbol{F^*} = \underset{\boldsymbol{F'} \subset \boldsymbol{F},\, card(\boldsymbol{F'}) = m}{arg\,max} \{J(\boldsymbol{F'})\}$$

where *J(·)* is an opportune function that measures the separability between the classes in the space defined by the considered subset of features.

Feature Selection

- Feature selection involves two important ingredients:

  ☞ a separability measure *J(·)*

  ☞ a search strategy in the solution space

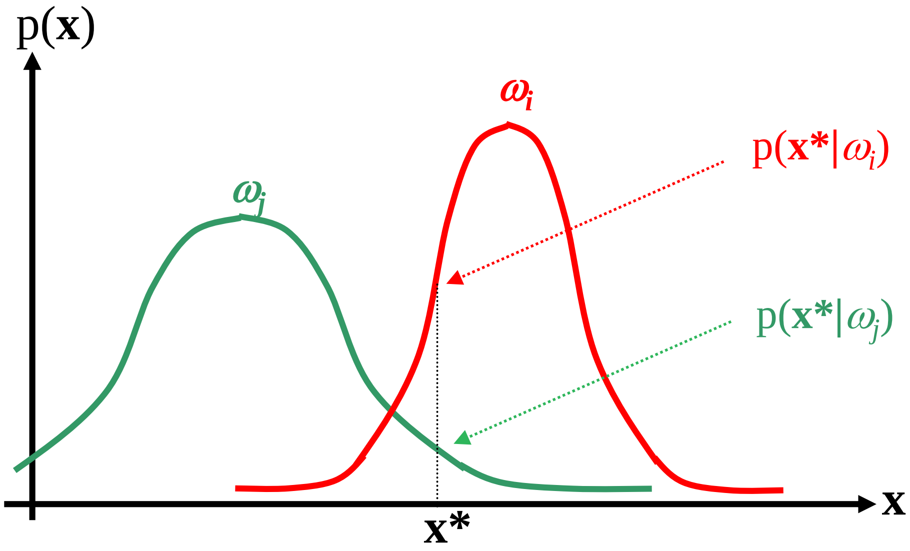
Feature Selection: Separability Measure

- The best separability measure should be compatible with the discrimination criterion adopted by the considered classifier.
- A trivial separability measure: accuracy achieved by the adopted classifier.
- Alternative: measure based on the error probability of the Bayes classifier.
- Typical separability measures used in feature selection are:
  - Divergence measure
  - Bhattacharyya distance
  - Jeffries-Matusita distance

Divergence Measure

- Let us consider a classification problem with two classes ω_i and ω_j characterized by the prior probabilities $P(\omega_i)$ and $P(\omega_j)$ and the class-conditional densities $p(\mathbf{x}|\omega_i)$ and $p(\mathbf{x}|\omega_j)$, respectively.
- **Underlying Idea**: measure the separability between the two classes in a considered feature subspace by computing the overlap degree between the two related densities.
- In order to define the divergence measure, let us first introduce the so-called likelihood ratio:

$$L_{ij}(\mathbf{x}) = \frac{p(\mathbf{x}\,|\,\omega_i)}{p(\mathbf{x}\,|\,\omega_j)}$$



Divergence Measure

- The divergence measure between the distributions of the two classes is defined as follows:

$$\mathrm{D}_{\mathrm{ij}}(\boldsymbol{F}') = E\{\mathrm{L}'_{\mathrm{ij}}(\mathbf{x})\} + E\{\mathrm{L}'_{\mathrm{ji}}(\overset{\mathbf{x}\in \boldsymbol{F}'}{\overset{\downarrow}{\enclose{circle}{\mathbf{x}}}})\}$$

where:

$$\mathrm{L}'_{\mathrm{ij}}(\mathbf{x}) = \ln\left[\mathrm{L}_{\mathrm{ij}}(\mathbf{x})\right] = \ln\left[\mathrm{p}(\mathbf{x}\,|\,\omega_i)\right] - \ln\left[\mathrm{p}(\mathbf{x}\,|\,\omega_j)\right]$$

- It can be easily verified that:

$$\mathrm{D}_{\mathrm{ij}}(\boldsymbol{F}') = \int_{\mathbf{x}} \left\{ \left[\mathrm{p}(\mathbf{x}\,|\,\omega_i) - \mathrm{p}(\mathbf{x}\,|\,\omega_j)\right] \ln\left[\frac{\mathrm{p}(\mathbf{x}\,|\,\omega_i)}{\mathrm{p}(\mathbf{x}\,|\,\omega_j)}\right] \right\} d\mathbf{x}$$

Divergence Measure

- If the class distributions are Gaussian, it can be shown that:

$$D_{ij}(\boldsymbol{F}') = \frac{1}{2}\mathrm{Tr}\left\{\left(\Sigma_i - \Sigma_j\right)\cdot\left(\Sigma_i^{-1} - \Sigma_j^{-1}\right)\right\} +$$

$$+\frac{1}{2}\mathrm{Tr}\left\{\left(\Sigma_i^{-1} - \Sigma_j^{-1}\right)\cdot\left(\mathbf{m}_i - \mathbf{m}_j\right)\cdot\left(\mathbf{m}_i - \mathbf{m}_j\right)^t\right\}$$

where Σ_i, Σ_j and $\mathbf{m_i}$, $\mathbf{m_j}$ are the covariance matrices and the mean vectors of the classes ω_i and ω_j, respectively. Tr{·} is the matrix trace operator.

Divergence Measure: Properties

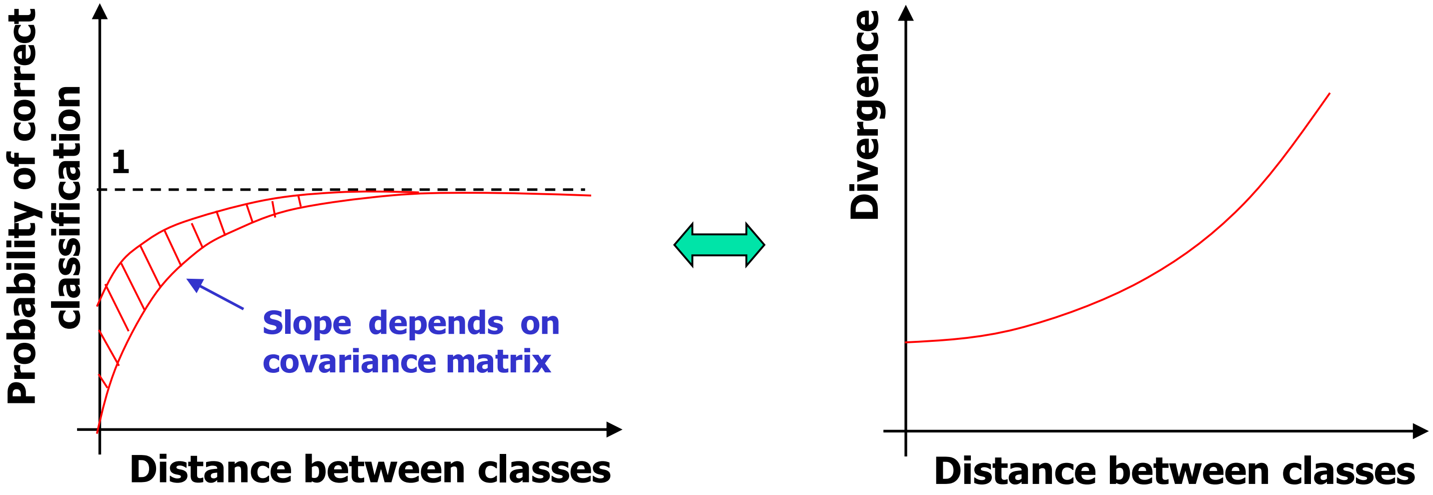
- $\omega_i = \omega_j \quad \Rightarrow \quad \mathrm{D_{ij}} = 0$
- $\omega_i \neq \omega_j \quad \Rightarrow \quad \mathrm{D_{ij}} > 0$
- $\mathrm{D_{ij}} = \mathrm{D_{ji}}$
- $\mathrm{D_{ij}}(f_1,.......,f_k) \leq \mathrm{D_{ij}}(f_1,.......,f_k, f_{k+1})$
- If the features are independent:

$$\mathrm{D_{ij}}(f_1,.......,f_k) = \sum_{q=1}^{k} \mathrm{D_{ij}}(f_q)$$

- The larger the divergence, the better the separability between classes.

Divergence Measure: Properties

- A drawback of the divergence measure is its non-saturating behavior as the distance between classes increases:



Divergence Measure: Multiclass Case

- What has been seen up to now holds for binary classification problems.
- For multiclass problems, a multiclass divergence measure could be deduced by:
  - Averaging the “binary” divergence measures corresponding to all couples of classes:

Number of classes

$$D_{ave}(\boldsymbol{F'}) = \sum_{i=1}^{C} \sum_{j>i}^{C} P(\omega_i) \cdot P(\omega_j) \cdot D_{ij}(\boldsymbol{F'})$$

  - Adopting the worst case reasoning, i.e., by using the lowest binary divergence measure:

$$D_{min}(\boldsymbol{F'}) = \min_{\substack{1 \le i \le C \\ i < j}} \{D_{ij}(\boldsymbol{F'})\}$$

Bhattacharyya Distance

- **Objective**: definition of a distance measure that depends analytically on the probability of error of the Bayes classifier.
- The Bhattacharyya distance is defined on the basis of an upper bound of such error probability.
- In order to define it, let us introduce the Chernoff bound.
- The probability of error P_e of the Bayes classifier is given by:

$$P_e = \int_{\mathbf{x}} \left\{ \min \left[P(\omega_i) p(\mathbf{x} \mid \omega_i) , P(\omega_j) p(\mathbf{x} \mid \omega_j) \right] \right\} d\mathbf{x}$$

- The direct computation of such quantity is not trivial. To overcome this issue, approximations are necessary.

Bhattacharyya Distance

- Using the following relationship:

$$\min[a\,,b] \le a^s b^{1-s}$$, where $0 \le s \le 1$

we can deduce an upper bound ε_u for the Bayes error, called Chernoff bound:

$$\varepsilon_u = P(\omega_i)^s P(\omega_j)^{1-s} \int_{\mathbf{x}} p(\mathbf{x}\,|\,\omega_i)^s \cdot p(\mathbf{x}\,|\,\omega_j)^{1-s}\, d\mathbf{x} = P(\omega_i)^s P(\omega_j)^{1-s} \exp\left[-\mu_{ij}(s)\right]$$

- The smallest the value of ε_u, the better the separability between ω_i and ω_j .
- The quantity $\mu_{ij}(s)$ is called the Chernoff distance.

Bhattacharyya Distance

- In the case of Gaussian class distributions, it can be shown that:

$$\mu_{ij}(s) = \frac{s(1-s)}{2}(\mathbf{m_i} - \mathbf{m_j})^t \left\{ s\Sigma_i + (1-s)\Sigma_j \right\}^{-1} (\mathbf{m_i} - \mathbf{m_j}) + \frac{1}{2} \ln \left\{ \frac{\left| s\Sigma_i + (1-s)\Sigma_j \right|}{|\Sigma_i|^s |\Sigma_j|^{1-s}} \right\}$$

- The Chernoff distance raises the problem of the estimation of the best value of s.
- This can be done empirically so that to maximize $\mu_{ij}(s)$.
- An alternative is to fix arbitrarily the value of s.
- A particular case is that corresponding to s=1/2. The resulting bound is called Bhattacharyya bound:

$$\varepsilon_u = \sqrt{P(\omega_i)P(\omega_j)} \int_{\mathbf{x}} \sqrt{p(\mathbf{x}|\omega_i) \cdot p(\mathbf{x}|\omega_j)}\, d\mathbf{x} = \sqrt{P(\omega_i)P(\omega_j)} \exp\left[-\mu_{ij}(1/2) \right]$$

Bhattacharyya Distance

- In a similar way, the Bhattacharyya distance can be deduced from the Chernoff distance by setting s=1/2.
- For Gaussian classes, the expression of the Bhattacharyya distance is:

$$B_{ij} = \mu_{ij}(1/2) = \frac{1}{8}(\mathbf{m_i} - \mathbf{m_j})^t \left\{ \frac{\Sigma_i + \Sigma_j}{2} \right\}^{-1} (\mathbf{m_i} - \mathbf{m_j}) + \frac{1}{2} \ln \left\{ \frac{\left| \frac{\Sigma_i + \Sigma_j}{2} \right|}{|\Sigma_i|^{1/2} |\Sigma_j|^{1/2}} \right\}$$

Bhattacharyya Distance: Properties

- $\omega_\mathrm{i} = \omega_\mathrm{j} \quad \Rightarrow \quad \mathrm{B}_\mathrm{ij} = 0$
- $\omega_\mathrm{i} \neq \omega_\mathrm{j} \quad \Rightarrow \quad \mathrm{B}_\mathrm{ij} > 0$
- $\mathrm{B}_\mathrm{ij} = \mathrm{B}_\mathrm{ji}$
- $\mathrm{B}_\mathrm{ij}(f_1,.......,f_k) \leq \mathrm{B}_\mathrm{ij}(f_1,.......,f_k,f_{k+1})$
- In case of independent features:

$$\mathrm{B}_\mathrm{ij}(f_1,......., f_k) = \sum_{q=1}^{k} \mathrm{B}_\mathrm{ij}(f_q)$$

- No saturating behavior
- Multiclass expression obtainable as done for the divergence distance

Jeffries-Matusita Distance

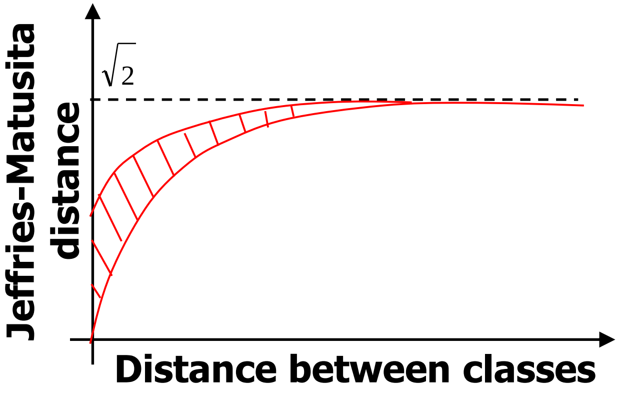
- **Objective**: definition of a distance measure with saturating behavior.

- The Jeffries-Matusita distance is an average distance between two density functions:

$$\mathrm{JM}_{\mathrm{ij}} = \left\{ \int_{\mathbf{x}} \left[\sqrt{\mathrm{p}(\mathbf{x} \mid \omega_i)} - \sqrt{\mathrm{p}(\mathbf{x} \mid \omega_j)} \right]^2 \mathrm{d}\mathbf{x} \right\}^{1/2}$$

- It can be shown that the Jeffries-Matusita distance can be expressed as a function of the Bhattacharyya distance:

$$\mathrm{JM}_{\mathrm{ij}} = \sqrt{2(1 - \exp(-\mathrm{B}_{\mathrm{ij}}))}$$



Search Strategies

- Once the separability criterion is adopted, it is necessary to resort to a search strategy in order to identify the best subset of features that optimizes the criterion.
- The most intuitive strategy is that based on an exhaustive search.
- It has the advantage to find the optimal solution but often requires a prohibitive computational cost.
- Indeed, for a set of *n* features, the number of subsets of *m* features *(m < n),* which should be explored, is given by the following binomial coefficient:

$$\binom{n}{m} = \frac{n!}{m!(n-m)!}$$

Search Strategies

- **Examples:**
  - $n = 10$

$$\begin{cases} \binom{10}{2} = \dfrac{10!}{2!8!} = 45 \quad (m=2) \\ \binom{10}{4} = \dfrac{10!}{4!6!} = 210 \quad (m=4) \end{cases}$$

  - $n = 200$

$$\binom{200}{30} = \frac{200!}{30!170!} = 4{,}09 \cdot 10^{35} \quad (m=30)$$

Search Strategies

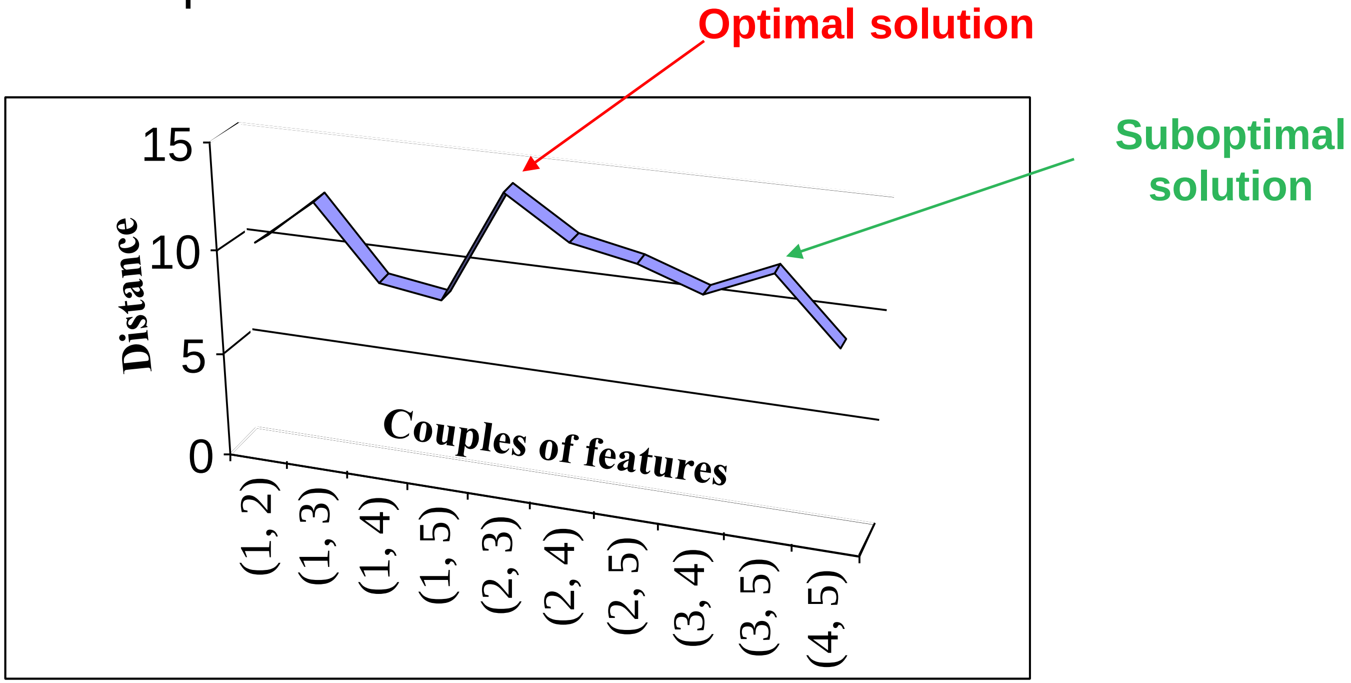
- Suboptimal strategies: for saving computational time, they explore just partially the solution space and thus lead to suboptimal solutions.

- Optimal strategies: they guarantee a convergence to the optimal solution but are more computationally demanding.

- In the following, first, we will see a popular suboptimal feature selection method.

Suboptimal Search Strategies

- **Example**: Selection of 2 features among 5 for a binary classification problem

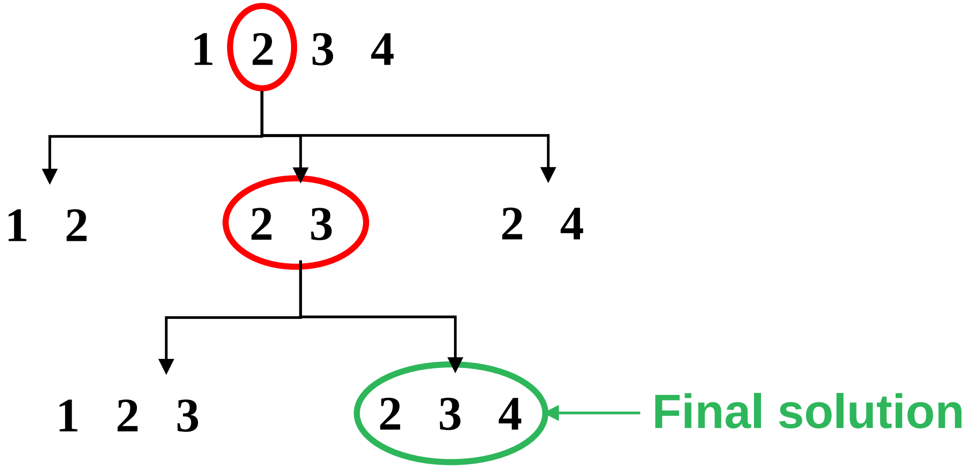


Sequential Forward Selection

- SFS is an iterative “bottom-up” search strategy, which consists first to find the feature that optimizes singularly the adopted class separability measure.
- This feature is the first one included in the desired subset of features.
- At each iteration, the feature which together with those selected in the previous iterations optimizes the adopted criterion, is also selected and added in the subset.
- This process is iterated up to select the desired number m of features.
- SFS is suboptimal because of the nesting effect, i.e., a feature previously selected can no more be removed from the subset.

Sequential Forward Selection

- Example: $m=3\ ;\ n=4$

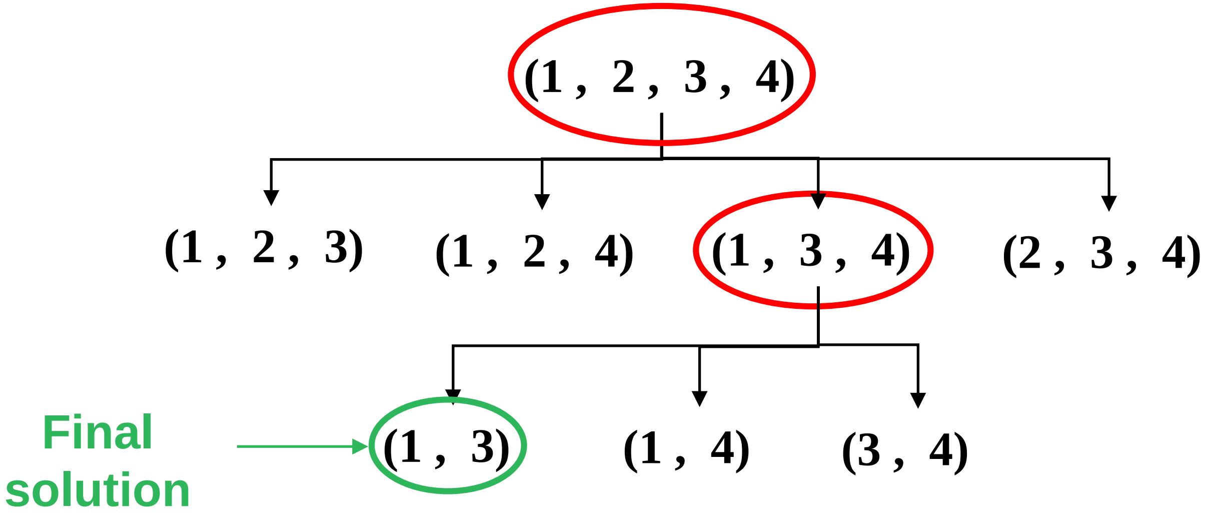


Sequential Backward Selection

- SBS is said to be a "top-down" search strategy since it works in the opposite direction with respect to that of SFS.
- At the beginning, all available features are included in the subset.
- At each iteration, the feature that involves the lowest decrease of the adopted class separability criterion is removed from the subset.
- This feature removal process continues up to get a subset with cardinality equal to *m*.

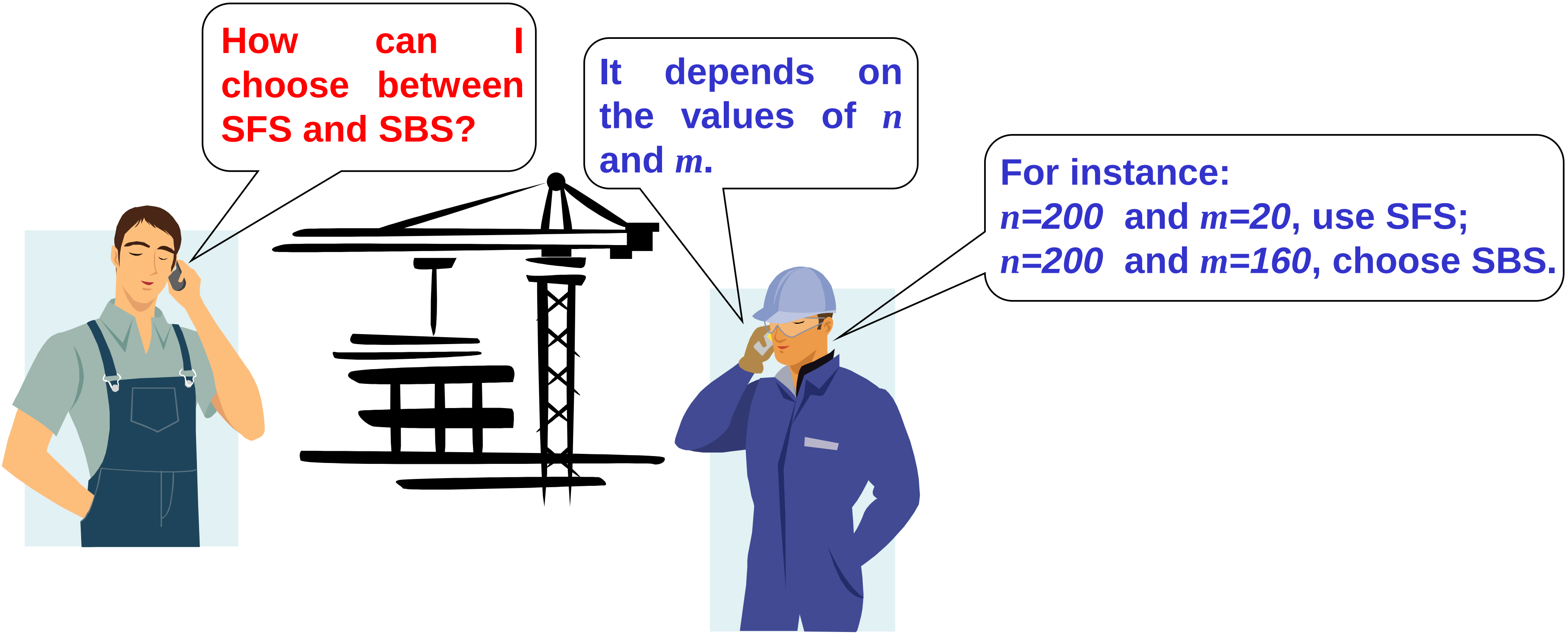
Sequential Backward Selection

- Example: $m=2 \; ; n=4$



SFS & SBS: Observations

- 👍 Very fast even with n >>.
- 👎 Feature insertion/removal process is irreversible.



How can I choose between SFS and SBS?

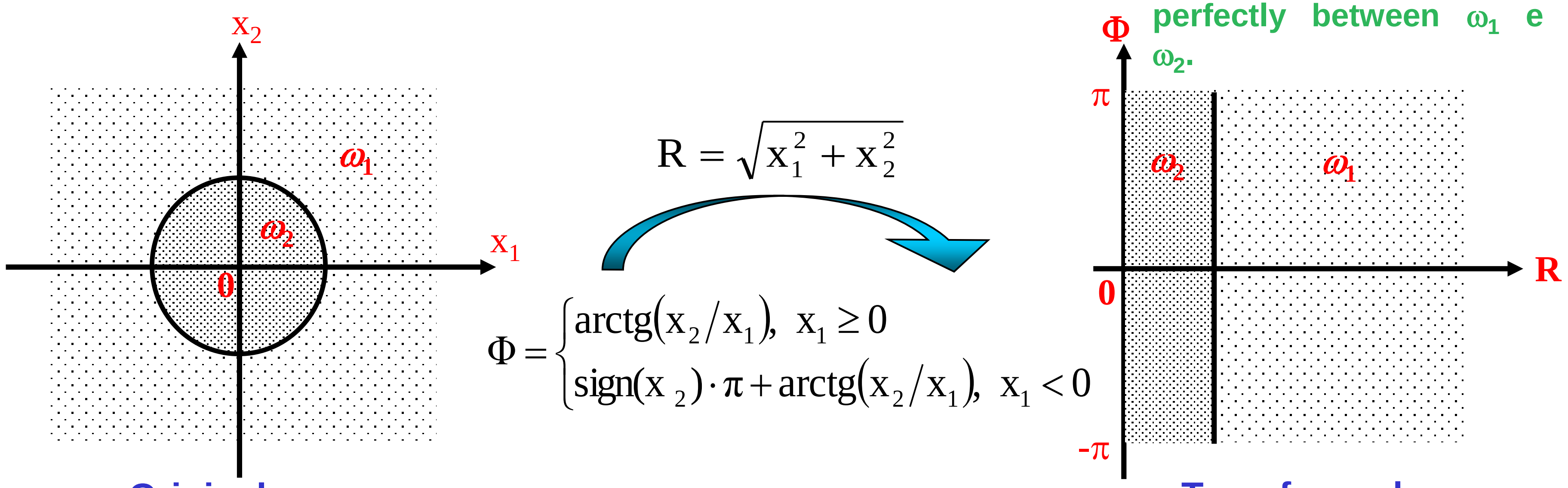
It depends on the values of n and m.

For instance:
n=200 and m=20, use SFS;
n=200 and m=160, choose SBS.

Feature Extraction

- An alternative approach for coping with the feature reduction problem is to combine features. It is often termed as feature extraction.
- The potential advantage of feature extraction is to lose less information than feature selection does for a fixed number of desired features.

The new feature R is enough to discriminate perfectly between ω_1 e ω_2.



$$R = \sqrt{x_1^2 + x_2^2}$$

$$\Phi = \begin{cases} \mathrm{arctg}(x_2/x_1), & x_1 \geq 0 \\ \mathrm{sign}(x_2)\cdot\pi + \mathrm{arctg}(x_2/x_1), & x_1 < 0 \end{cases}$$

Original space

Transformed space

Feature Extraction

- Issues in feature extraction:
  - Linear versus nonlinear transformations
  - Use of class labels or not
  - Training objective:
    - minimizing classification error (discriminative training)
    - maximizing class separability (linear discriminant analysis)
    - retaining interesting directions (projection pursuit)
    - minimizing reconstruction error (principal component analysis)
    - making features as independent as possible (independent component analysis)

Feature Extraction

- Linear combinations are particularly attractive because they are simple to compute and are analytically tractable.
- Linear methods project the high-dimensional data onto a lower dimensional space.
- Advantages of these projections include:
  - reduced complexity in estimation and classification
  - ability to visually examine the multivariate data in two or three dimensions.
- Given $\mathbf{x} \in \Re^n$, the goal is to find a linear transformation Φ such that:

$$\mathbf{y} = \Phi^t \mathbf{x} \in \Re^m \text{ where } m < n.$$

Feature Extraction

- Two classical approaches for finding optimal linear transformations are:

☞ **Principal Components Analysis (PCA): Seeks a projection that best represents the data in a least squares sense.**

☞ **Linear Discriminant Analysis (LDA): Seeks a projection that best separates the data in a least squares sense.**

Principal Components Analysis

- The Principal Component Analysis (PCA) or Karhunen-Loéve transform is an unsupervised feature extraction method frequently used in pattern recognition and signal/image processing applications.
- Given $\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N \in \Re^n$, the goal is to find a m-dimensional subspace where the reconstruction error of $\mathbf{x}_i$ in this subspace is minimized.
- The criterion function for the reconstruction error can be defined in the least-squares sense as:

$$J_m = \sum_{i=1}^{N} \left\| \sum_{k=1}^{m} \mathbf{y}_{ik}\phi_k - \mathbf{x}_i \right\|^2$$

where $\phi_1,\ldots,\phi_m$ are the bases for the subspace (stored as the columns of Φ) and $\mathbf{y}_i$ is the projection of $\mathbf{x}_i$ onto that subspace.

Principal Components Analysis

- It can be shown that J_m is minimized when $\phi_1, \ldots, \phi_m$ are the m eigenvectors of the scatter matrix

$$S = \sum_{i=1}^{N} (\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^{\mathrm{t}}$$

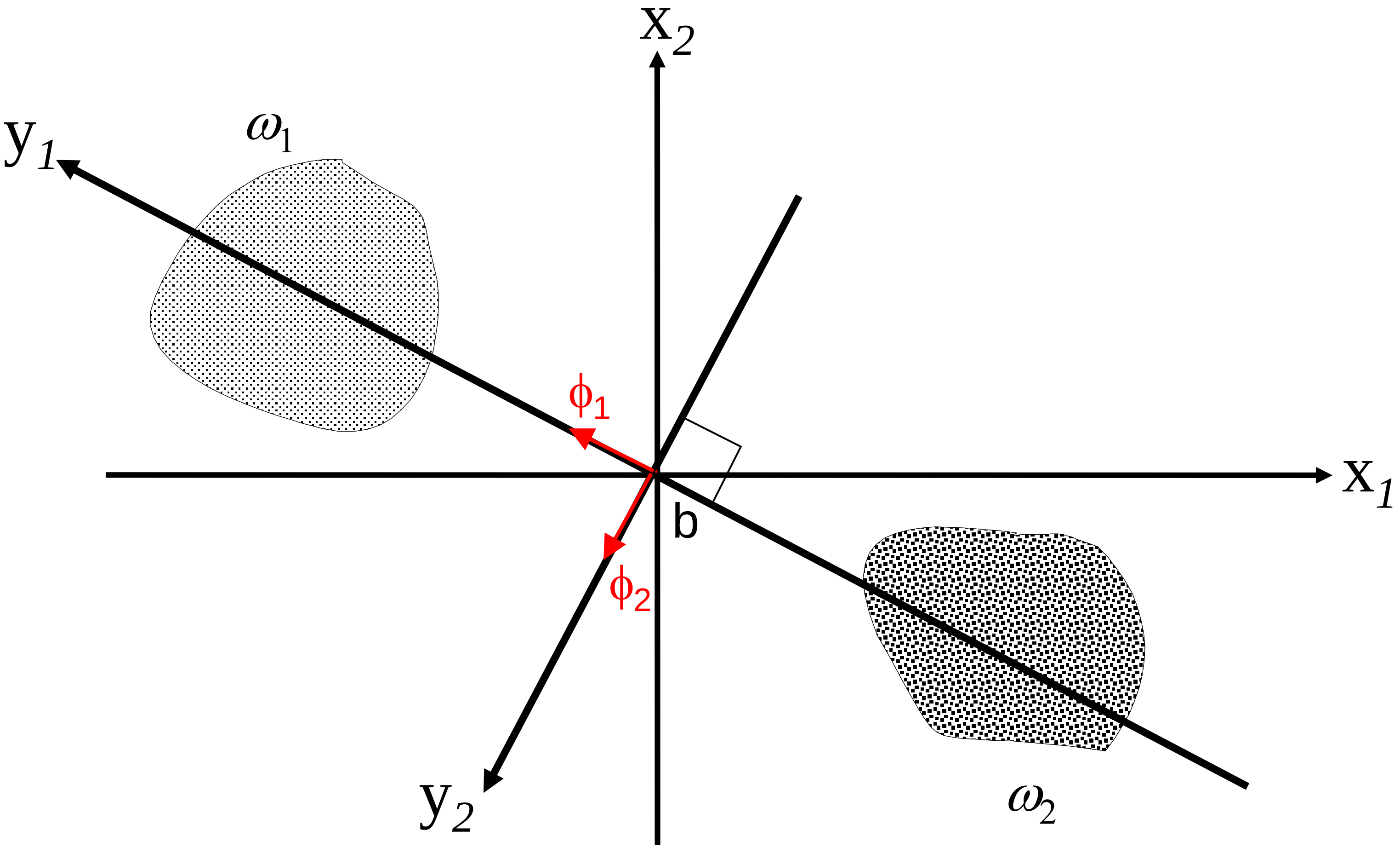
  having the largest eigenvalues.

Nota: *S* is *N*-1 times the covariance matrix Σ.

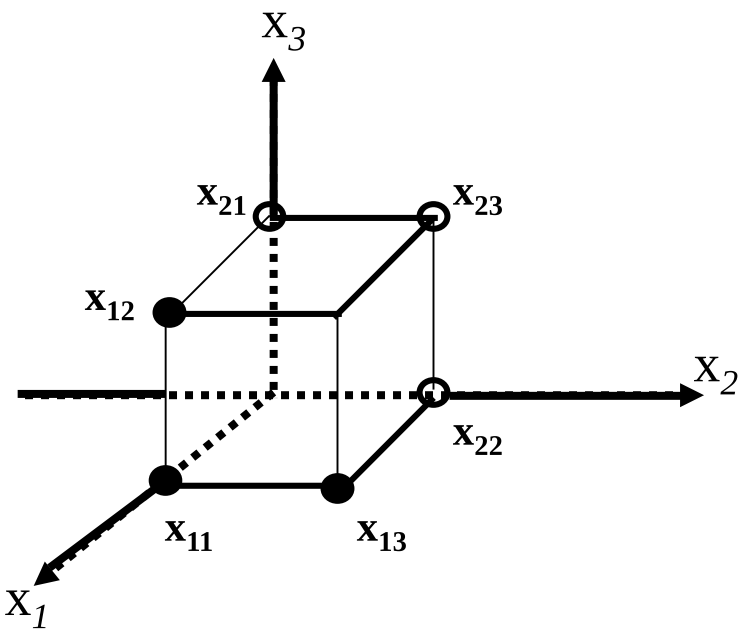
- The coefficients $\mathbf{y}$ = $(\mathrm{y}_1, \ldots, \mathrm{y}_\mathrm{m})^\mathrm{t}$ are called the principal components.
- When the eigenvectors are sorted in descending order of the corresponding eigenvalues, the greatest variance of the data lies on the first principal component, the second greatest variance on the second component, etc.
- Often there will be just a few large eigenvalues, and this implies that the m-dimensional subspace contains the signal and the remaining n-m dimensions generally contain noise.

Principal Components Analysis

- **Graphical illustration:**



PCA: Example



Class ω_1 (●)

$$\mathbf{x}_{11} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{x}_{12} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

$$\mathbf{x}_{13} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

Class ω_2 (○)

$$\mathbf{x}_{21} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\mathbf{x}_{22} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\mathbf{x}_{23} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

Objective: reduce number of features

- from 3 to 2

- from 3 to 1

PCA: Example

Step 1: Compute the barycenter

$$\mathbf{m} = \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \end{bmatrix}$$

Step 2: Data shifting $\mathbf{x}' = \mathbf{x} - \mathbf{m}$

$$\mathbf{x}'_{11} = \begin{bmatrix} 0.5 \\ -0.5 \\ -0.5 \end{bmatrix} \quad \mathbf{x}'_{12} = \begin{bmatrix} 0.5 \\ -0.5 \\ 0.5 \end{bmatrix} \quad \mathbf{x}'_{13} = \begin{bmatrix} 0.5 \\ 0.5 \\ -0.5 \end{bmatrix}$$

$$\mathbf{x}'_{21} = \begin{bmatrix} -0.5 \\ -0.5 \\ 0.5 \end{bmatrix} \quad \mathbf{x}'_{22} = \begin{bmatrix} -0.5 \\ 0.5 \\ -0.5 \end{bmatrix} \quad \mathbf{x}'_{23} = \begin{bmatrix} -0.5 \\ 0.5 \\ 0.5 \end{bmatrix}$$

PCA: Example

Step 3: Computation of covariance matrix Σ

$$\Sigma_{X'} = \frac{1}{12}\begin{bmatrix} 3 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 3 \end{bmatrix}$$

Step 4: Computation of eigenvalues and eigenvectors

$$\begin{cases} \lambda_1 = 1/3 \\ \lambda_2 = 1/3 \\ \lambda_3 = 1/12 \end{cases} \Rightarrow \quad \phi_1 = \begin{bmatrix} 0.79 \\ -0.23 \\ -0.56 \end{bmatrix} \quad \phi_2 = \begin{bmatrix} 0.19 \\ -0.78 \\ 0.59 \end{bmatrix} \quad \phi_3 = \begin{bmatrix} 0.58 \\ 0.58 \\ 0.58 \end{bmatrix}$$

PCA: Example

Step 5: Reduction to 2 features

- Form transformation matrix:

$$\Phi^t = [\phi_1\ \phi_2]^t = \begin{bmatrix} 0.79 & -0.23 & -0.56 \\ 0.19 & -0.78 & 0.59 \end{bmatrix}$$

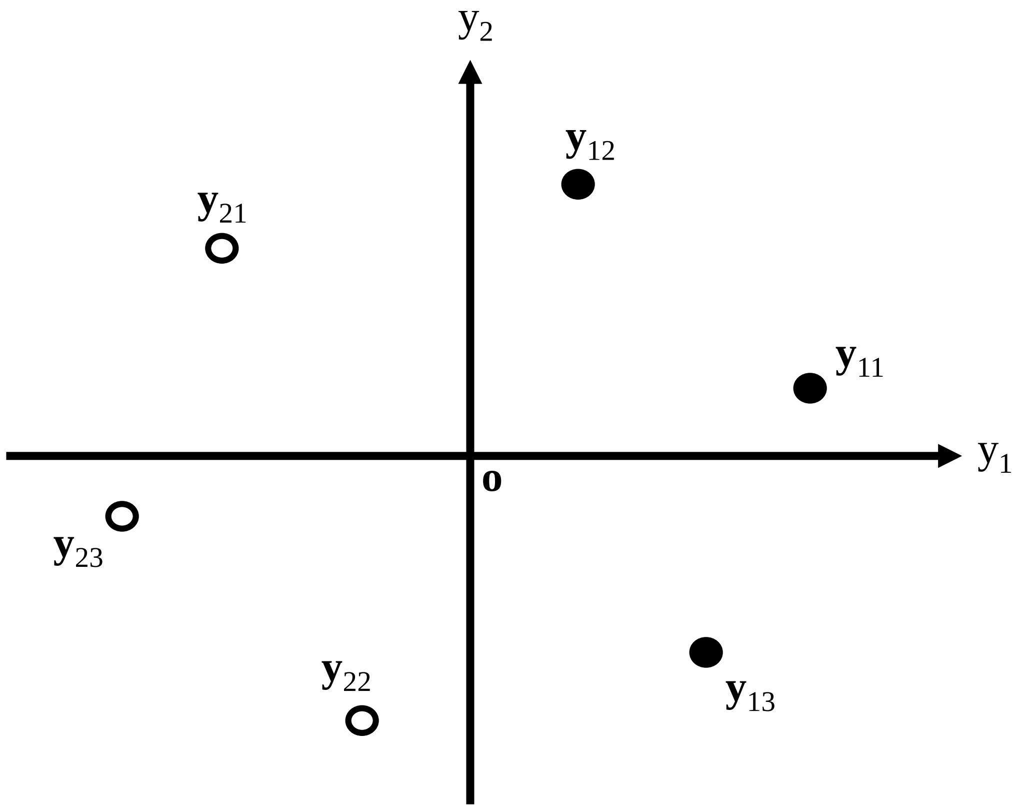
- Apply transformation $\mathbf{y} = \Phi^t\mathbf{x}$ to all samples:

$$\mathbf{y}_{11} = \begin{bmatrix} 0.79 \\ 0.19 \end{bmatrix} \qquad \mathbf{y}_{12} = \begin{bmatrix} 0.23 \\ 0.78 \end{bmatrix} \qquad \mathbf{y}_{13} = \begin{bmatrix} 0.56 \\ -0.59 \end{bmatrix}$$

$$\mathbf{y}_{21} = \begin{bmatrix} -0.56 \\ 0.59 \end{bmatrix} \qquad \mathbf{y}_{22} = \begin{bmatrix} -0.23 \\ -0.78 \end{bmatrix} \qquad \mathbf{y}_{23} = \begin{bmatrix} -0.79 \\ -0.19 \end{bmatrix}$$

PCA: Example

- Sample distribution in the first two components



PCA: Example

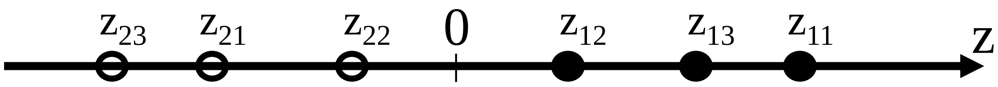
Step 5’: Reduction to 1 feature

- In this case, $\Phi = \phi_1$.
- Apply transformation $z = \Phi^t \mathbf{x}$ to all samples:

$$z_{11} = 0.79 \qquad z_{21} = -0.56$$

$$z_{12} = 0.23 \qquad z_{22} = -0.23$$

$$z_{13} = 0.56 \qquad z_{23} = -0.79$$



Linear Discriminant Analysis

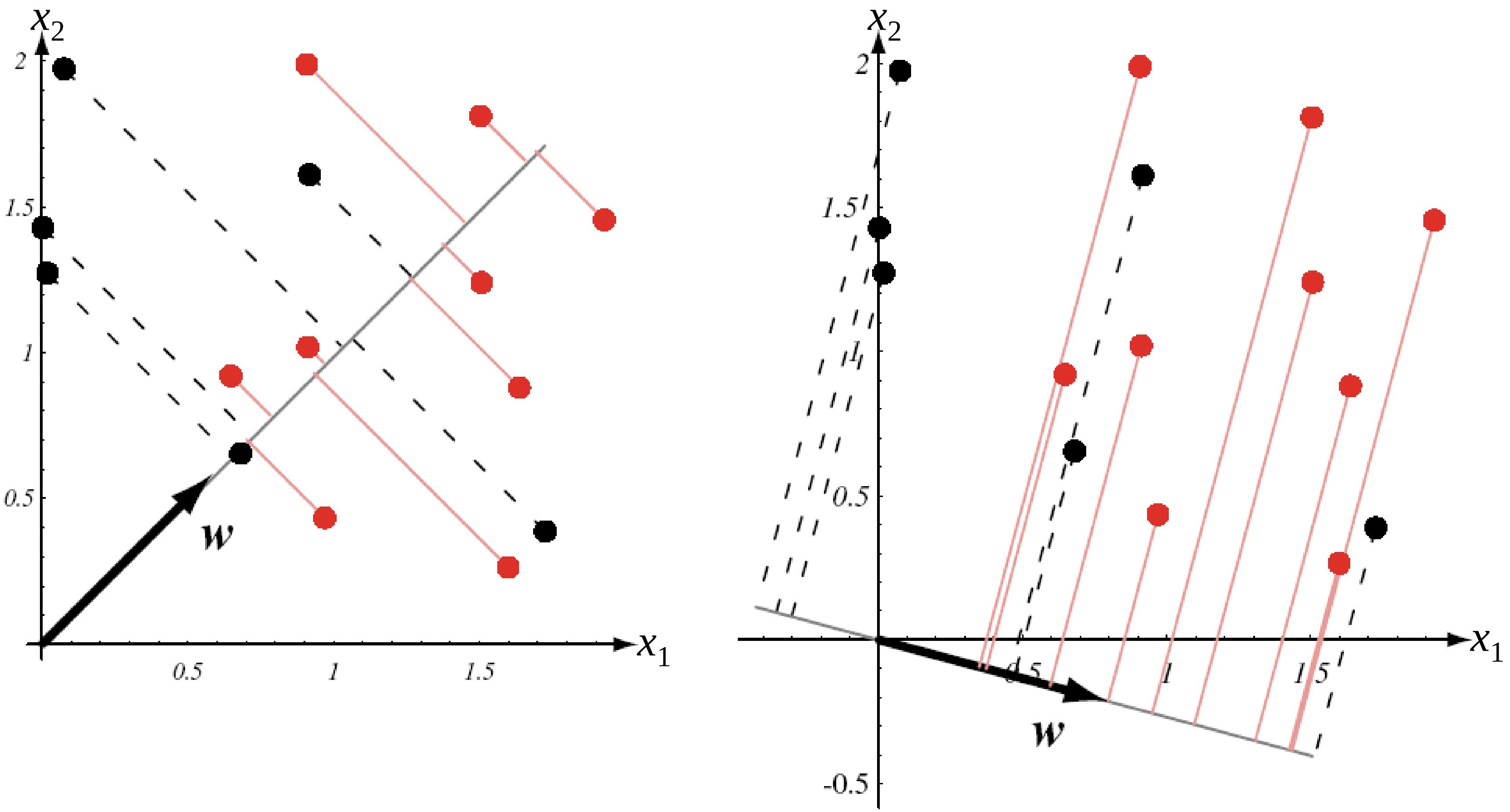
- Whereas PCA seeks directions that are efficient for representation, discriminant analysis seeks directions that are efficient for discrimination.
- Given $\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N \in \Re^n$ divided into two subsets $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ corresponding to the classes ω_1 and ω_2, respectively, the goal is to find a projection onto a line defined as

$$y = \mathbf{w}^t\mathbf{x}$$

where the points corresponding to $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ are well separated.

Linear Discriminant Analysis

Projection of the same set of samples onto two different lines in the directions marked w.



Linear Discriminant Analysis

- A criterion function for best separation could be defined as

$$J(\mathbf{w}) = \frac{\| \tilde{m}_1 - \tilde{m}_2 \|^2}{\tilde{s}_1^2 + \tilde{s}_2^2}$$

where $\tilde{m}_i = \frac{1}{\#(\boldsymbol{X}_i)} \sum_{y \in \omega_i} y$ is the sample mean and $\tilde{s}_i^2 = \sum_{y \in \omega_i} (y - \tilde{m}_i)^2$

is the scatter for the projected samples labeled ω_i.

- This is called the Fisher's linear discriminant with the geometric interpretation that the best projection makes the difference between the means as large as possible relative to the variance.

Linear Discriminant Analysis

- To compute the optimal $\mathbf{w}$, we define:
  - the scatter matrices S_i

$$S_i = \sum_{\mathbf{x} \in \boldsymbol{X}_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^{\mathrm{t}} \quad \text{where} \quad \mathbf{m}_i = \frac{}{\#(\boldsymbol{X}_i)} \sum_{\mathbf{x} \in \boldsymbol{X}_i} \mathbf{x}$$

  - the within-class scatter matrix S_w

$$S_W = S_1 + S_2$$

  - the between-class scatter matrix S_B

$$S_B = (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^{\mathrm{t}}$$

Linear Discriminant Analysis

- Then, the criterion function becomes:

$$J(\mathbf{w}) = \frac{\mathbf{w}^{\mathbf{t}} S_B \mathbf{w}}{\mathbf{w}^{\mathbf{t}} S_W \mathbf{w}}$$

  and the optimal $\mathbf{w}$ can be computed as

$$\mathbf{w} = S_W^{-1}(\mathbf{m}_1 - \mathbf{m}_2)$$

- Note that, S_w is symmetric and positive semidefinite, and it is usually nonsingular if $N > n$. S_B is also symmetric and positive semidefinite, but its rank is at most 1.

LDA: Multiclass Case

- Generalization to C classes involves C − 1 discriminant functions where the projection is from a n-dimensional space to a (C−1)-dimensional space ($n \geq C$).

- The within-class scatter matrix S_w becomes:

$$S_W = \sum_{i=1}^{C} S_i$$

- The between-class scatter matrix S_B is:

$$S_B = \sum_{i=1}^{C} [\#(\boldsymbol{X}_i)](\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^{\mathrm{t}}$$

Global mean vector

$$\mathbf{m} = \frac{1}{N} \sum_{\mathbf{x} \in \boldsymbol{X}} \mathbf{x}$$

LDA: Multiclass Case

- The criterion function is given by:

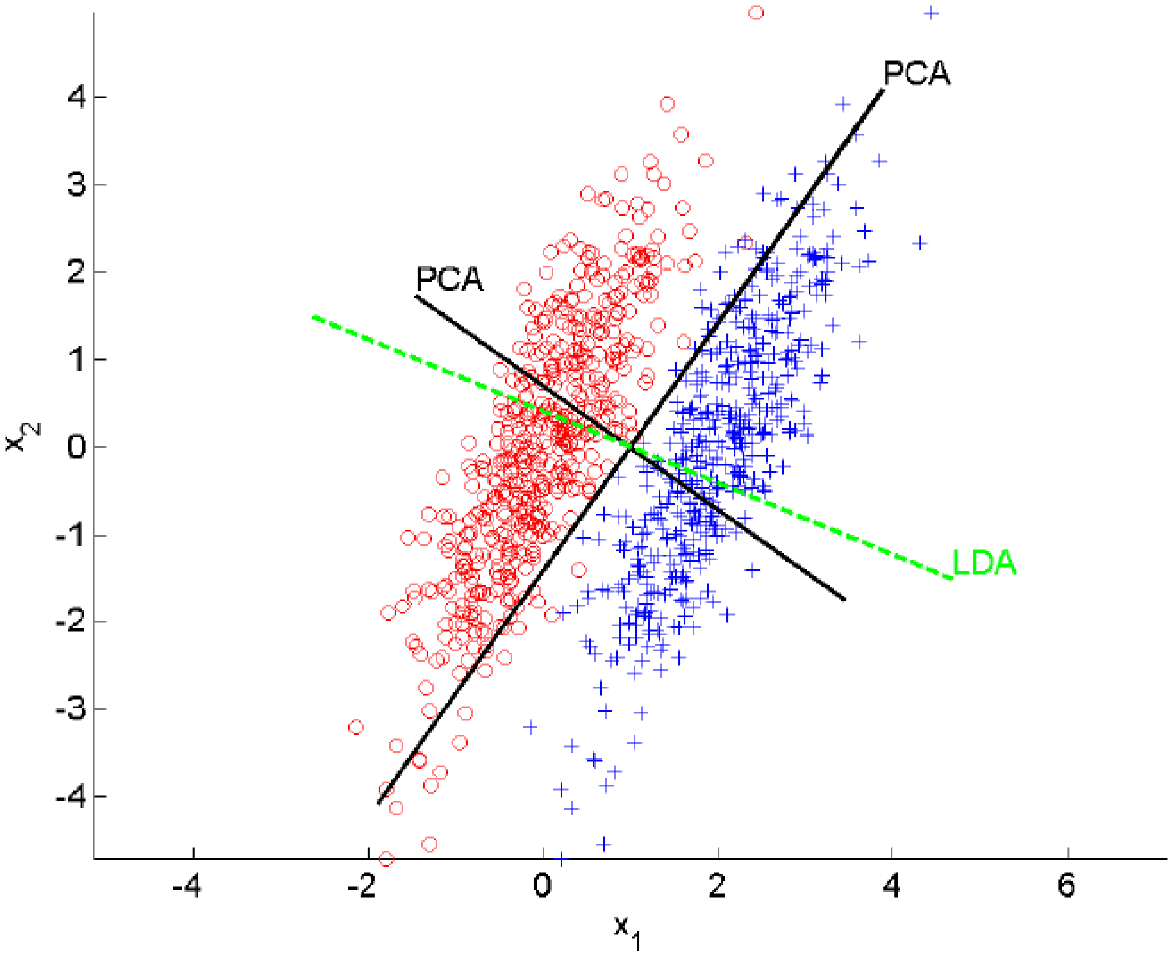
$$J(\mathbf{W}) = \frac{\left|\mathbf{W}^{\mathbf{t}} S_B \mathbf{W}\right|}{\left|\mathbf{W}^{\mathbf{t}} S_W \mathbf{W}\right|}$$

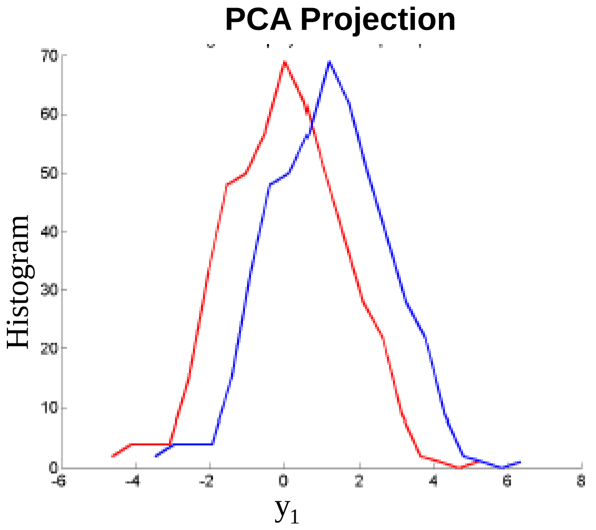
where $\mathbf{W}$ is the n–by-(C-1) transformation matrix and |·| represents the determinant.

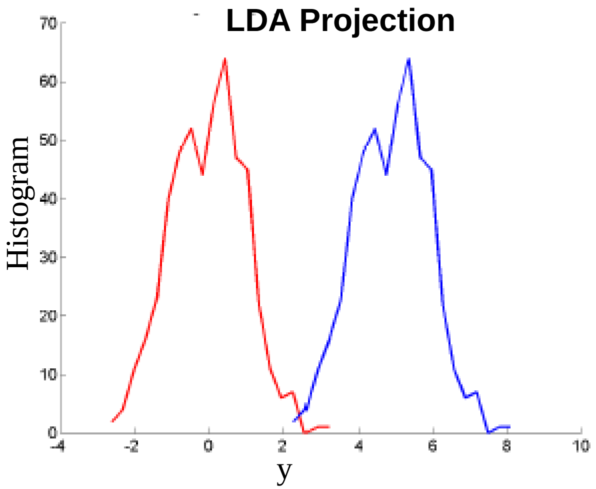
- It can be shown that $J(\mathbf{W})$ is maximized when the columns of $\mathbf{W}$ are the eigenvectors of $S_W^{-1} S_B$ having the largest eigenvalues.

- Because S_B is the sum of C matrices of rank one or less, and because only C−1 of these are independent, S_B is of rank C−1 or less. Thus, no more than C−1 of the eigenvalues are nonzero.

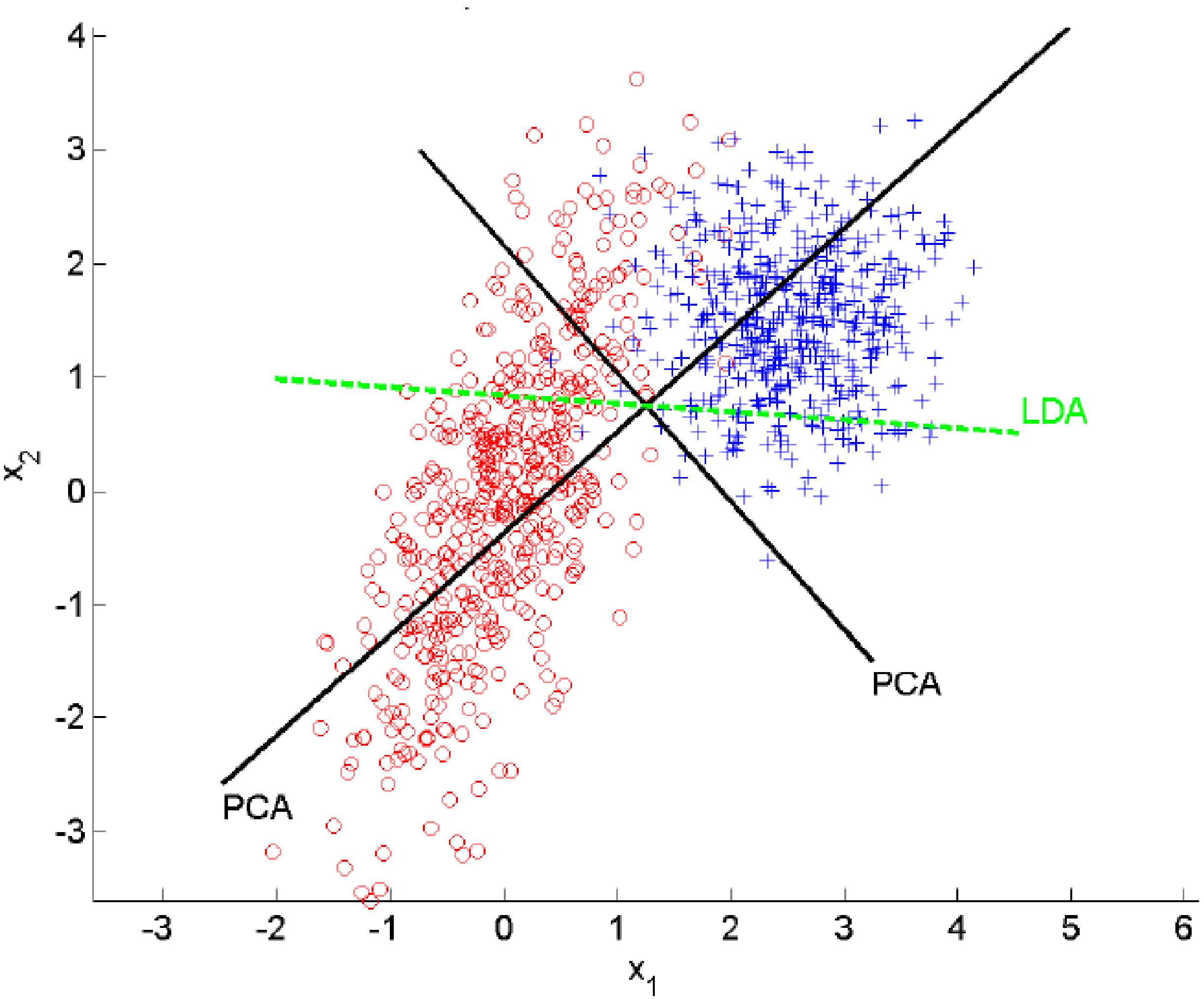
PCA Versus LDA: Example 1



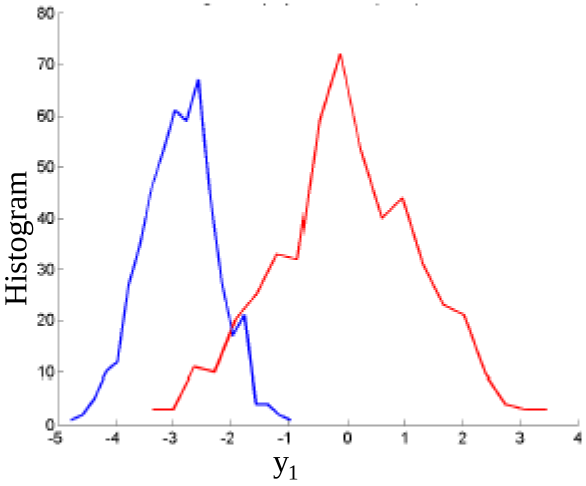




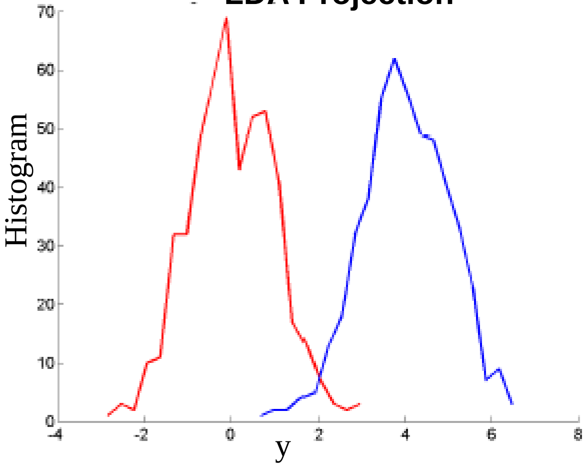
PCA Versus LDA: Example 2



PCA Projection

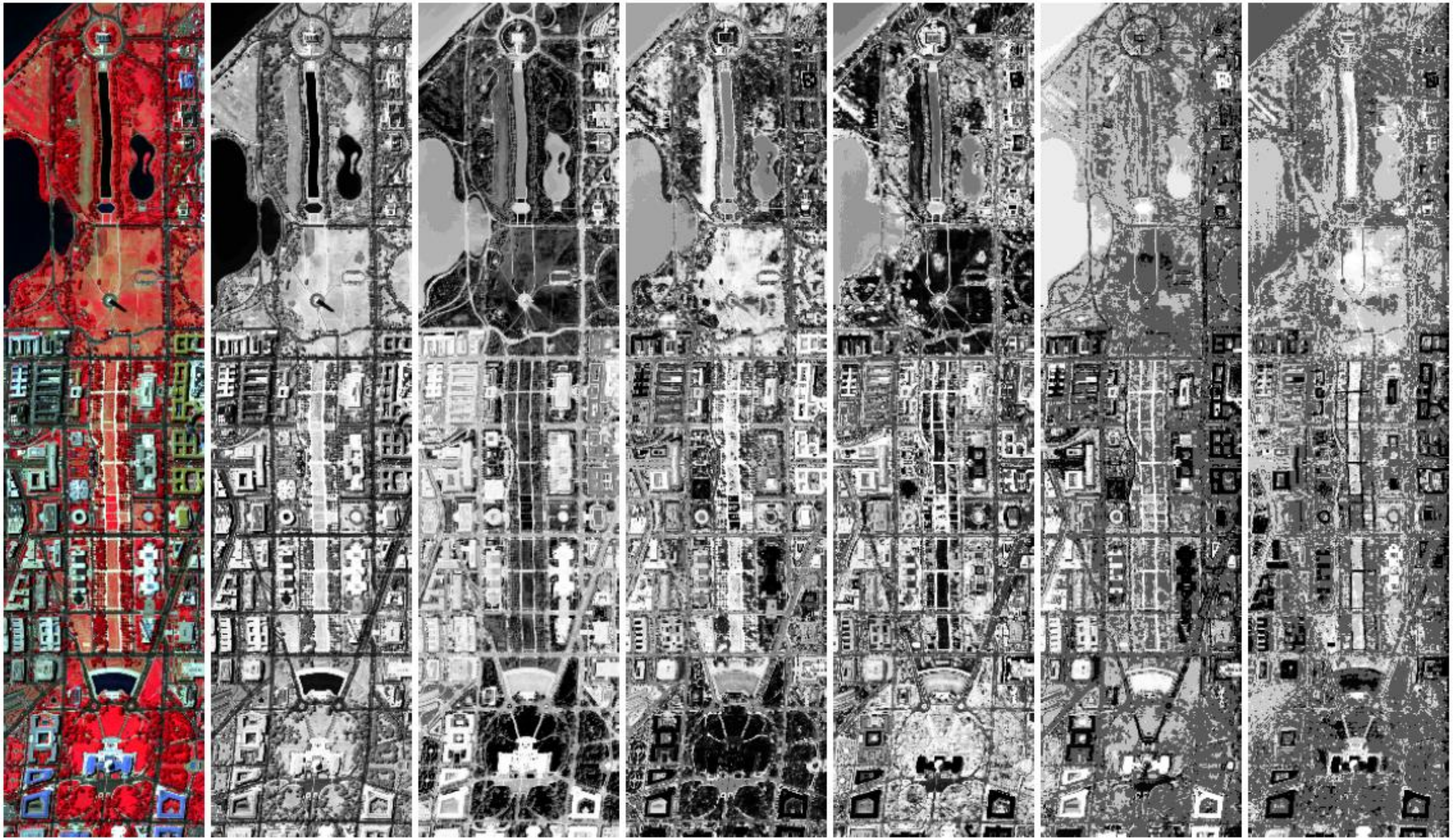


LDA Projection



PCA Versus LDA: Example 3

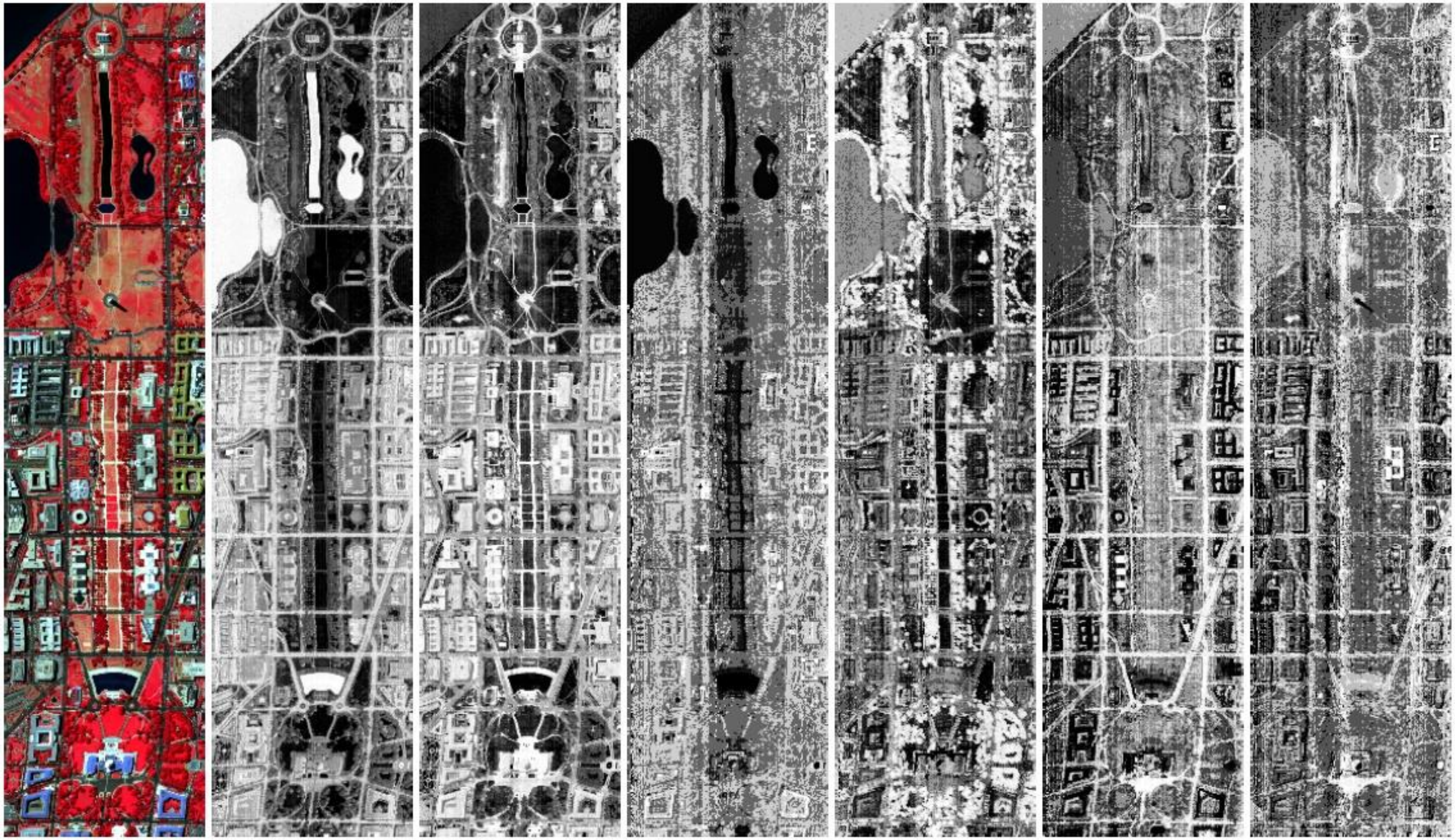
Remote sensing image and the first six PCA bands



Histogram equalization was applied to all images for better visualization.

PCA Versus LDA: Example 3

Remote sensing image and the six LDA bands



Histogram equalization was applied to all images for better visualization.