



模式识别 (9)

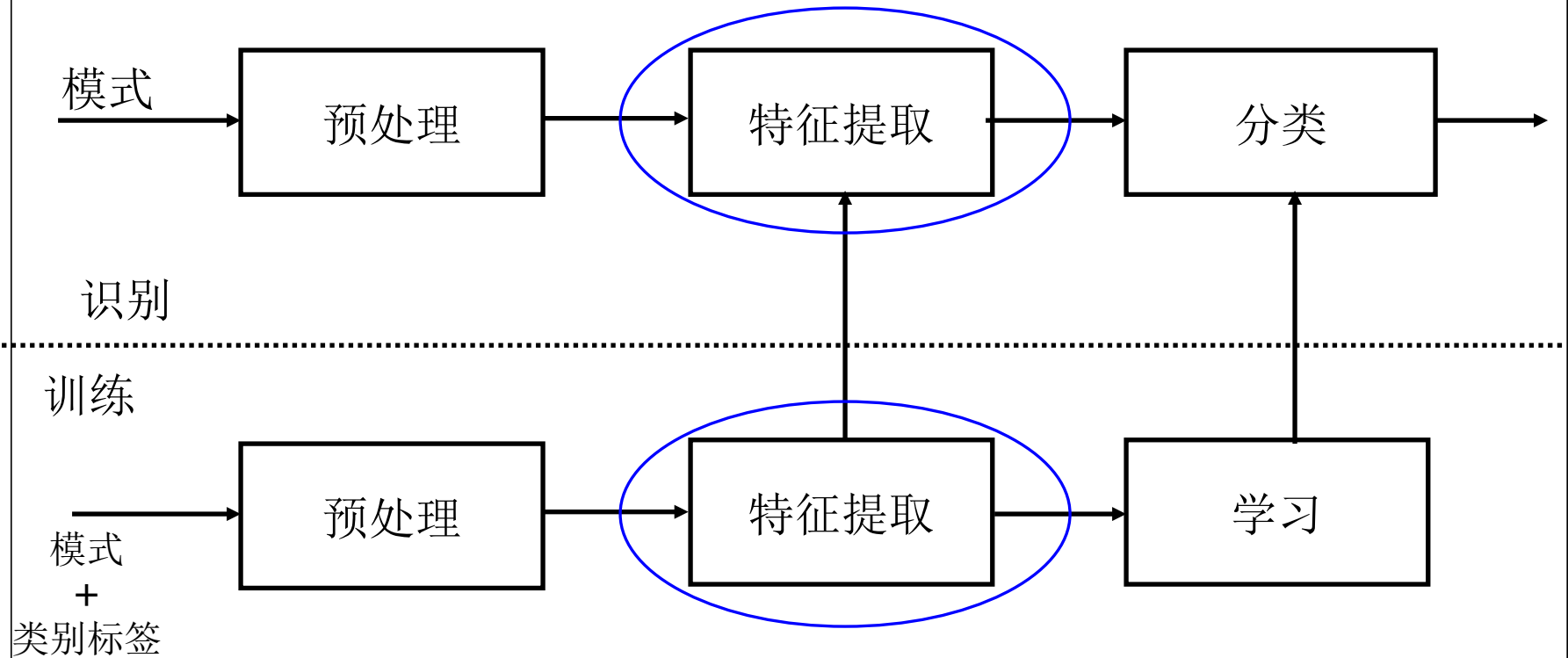
特征选择与特征提取

左旺孟

哈尔滨工业大学计算机学院
综合楼712

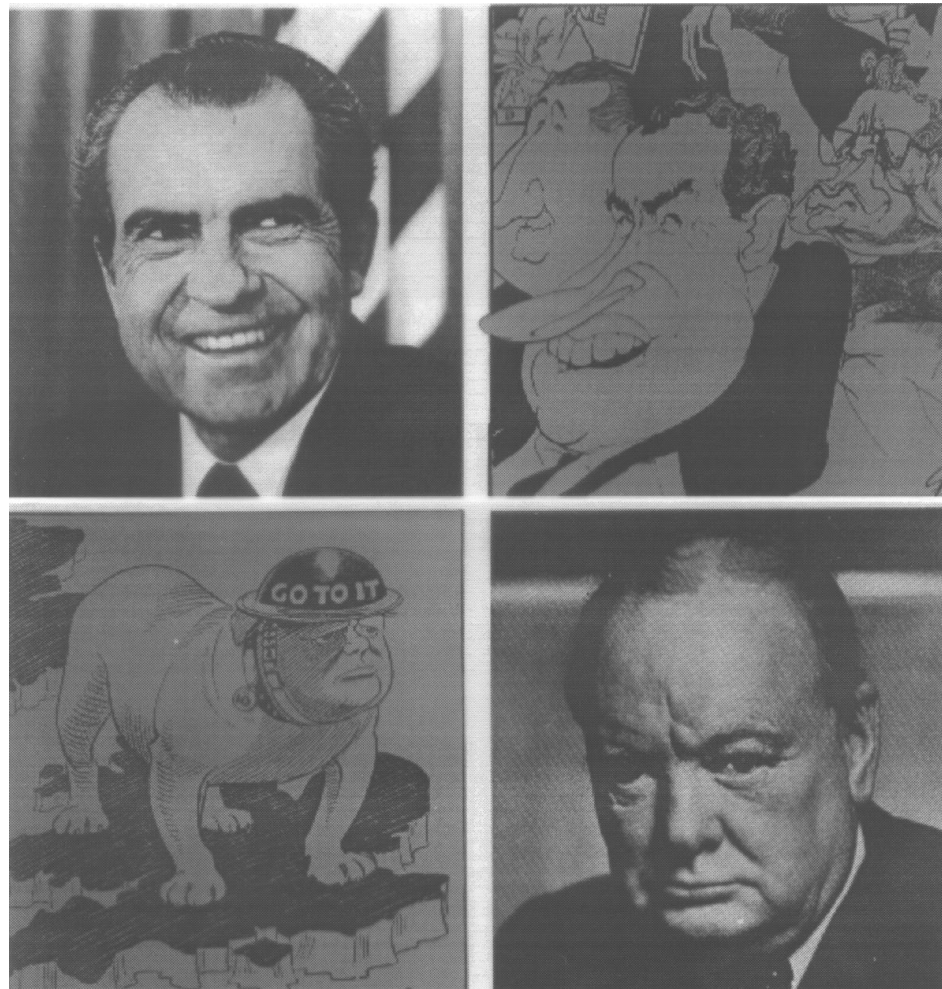
cswmzuo@gmail.com
13134506692

特征选择和特征提取



特征表示

- 类内变化
- 类间变化



特征表示



AMERICAN ELM



GINKGO



WEeping WILLOW



SPRUCE



LARCH



CANYON LIVE OAK



TELEPHONE POLE



BIRCH



MONTEREY CYPRESS



SCRUB PINE



DATE PALM



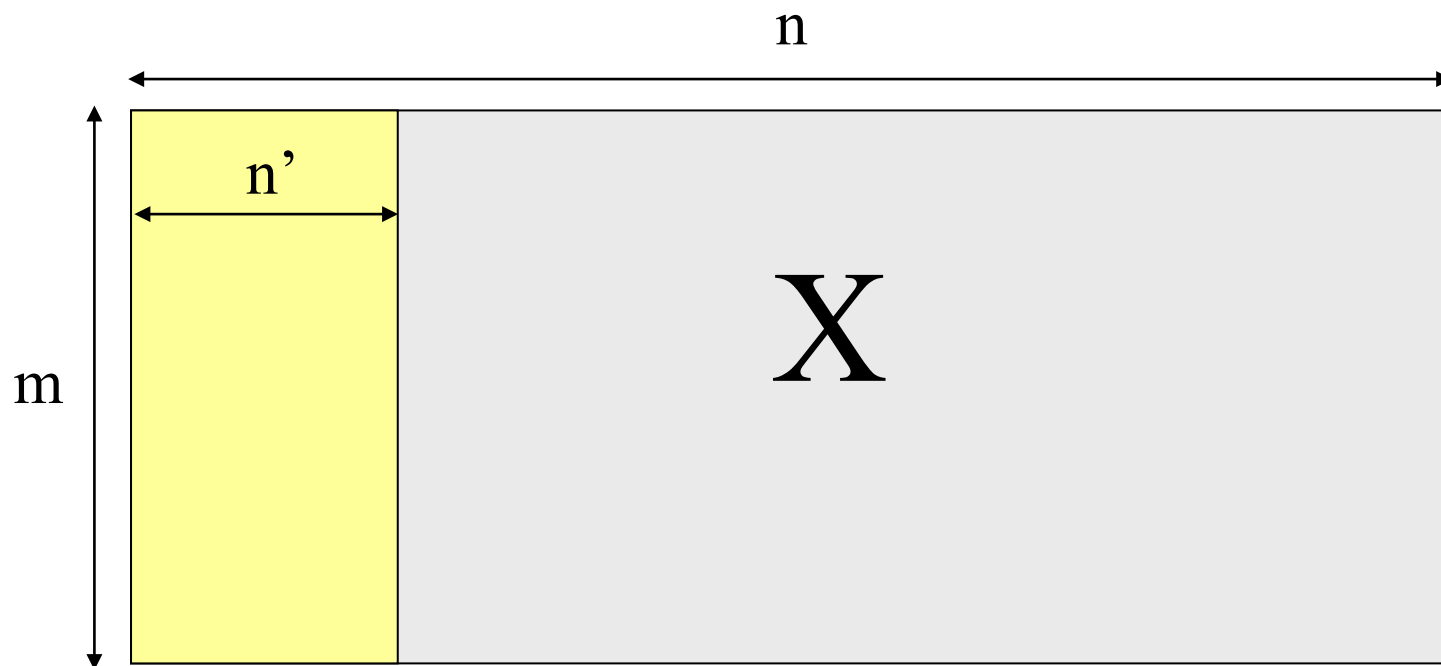
RED MANGROVE

主要内容

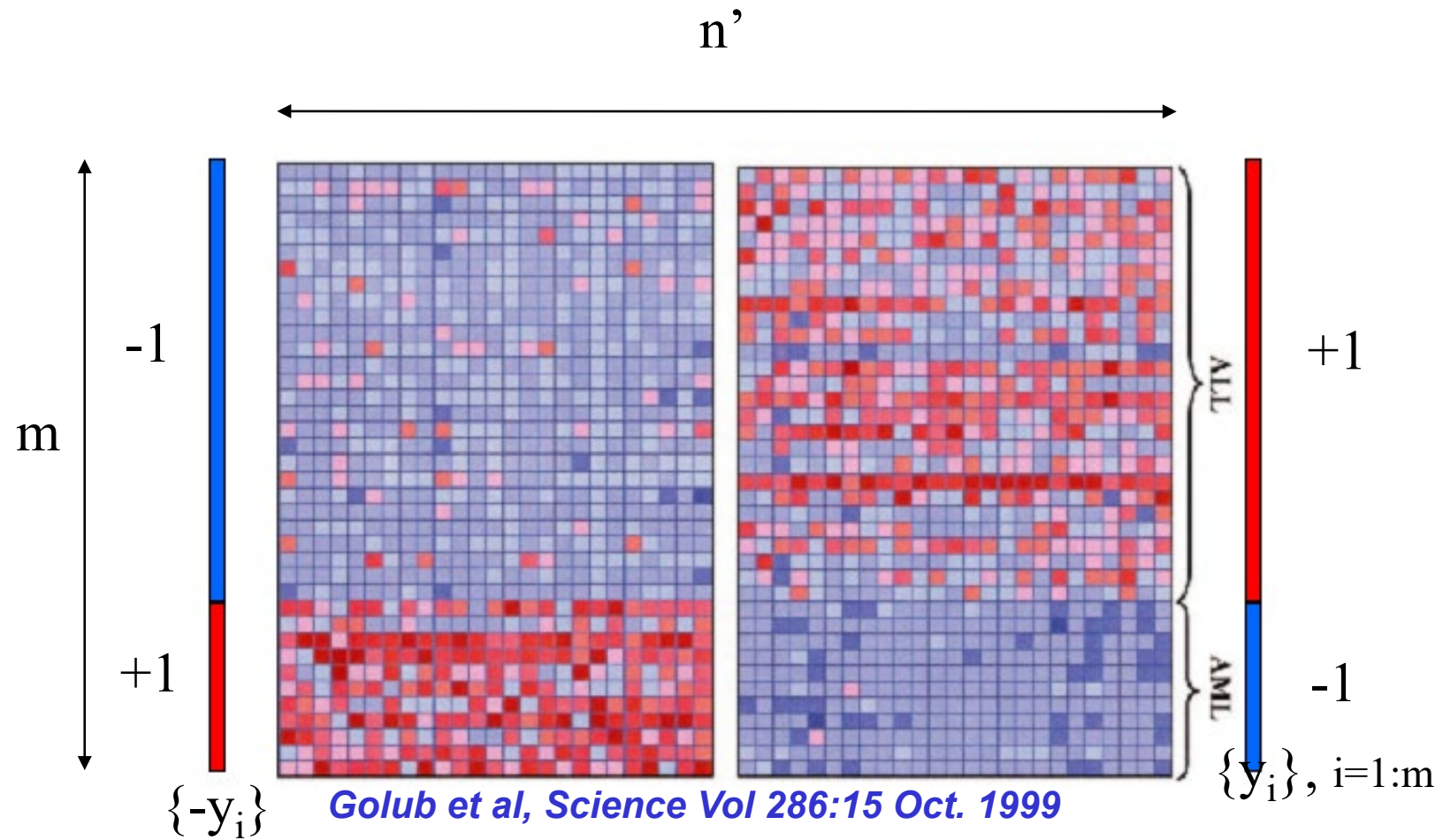
- 特征选择
 - 单变量选择法
 - 搜索法
 - 嵌入式方法
- 特征提取
 - 线性方法
 - 主成分分析（重点介绍）
 - 线性判别分析（重点介绍）
 - 非线性方法
 - 流形学习
 - 核特征提取方法

特征选择

- 去除冗余特征，提高识别性能
 - 冗余特征可能会导致性能恶化（维数灾难）
- 减少特征数目，提高识别速度
- 降低系统成本

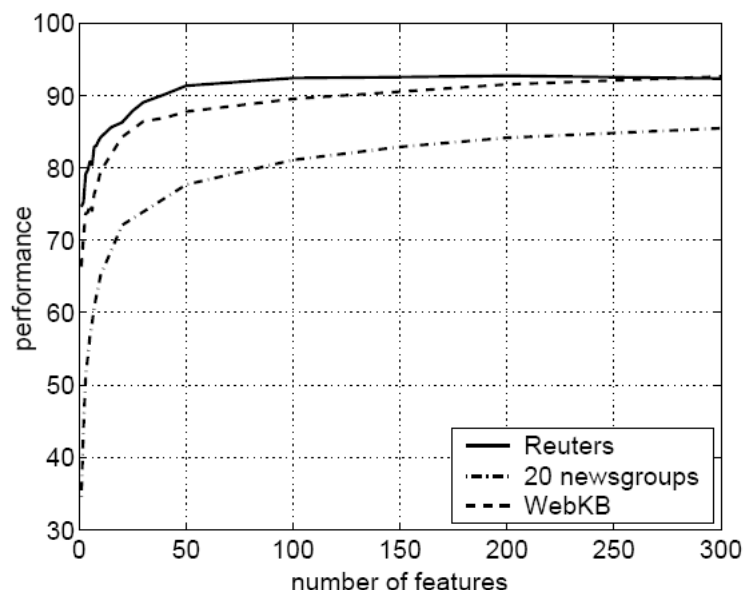


微阵列数据分析



- 发现相关致病基因

文本过滤



Reuters: 21578 新闻报道, 114 个语义类.

20 newsgroups: 19997 篇文章, 20 类.

WebKB: 8282个网页, 7 类.

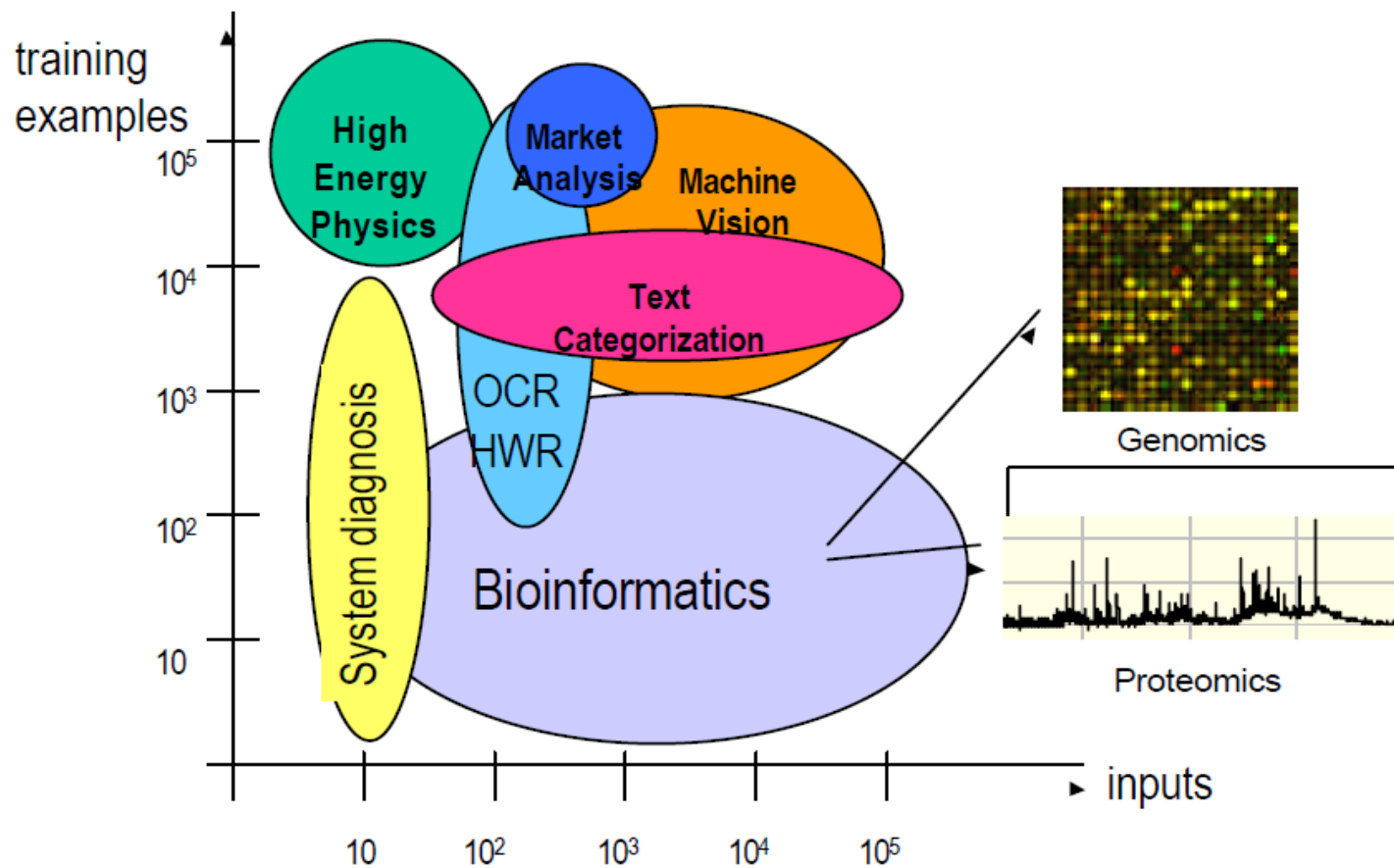
原始特征维数: >100000.

某些类的前三个特征词:

- ❖ **Alt.atheism:** atheism, atheists, morality
- ❖ **Comp.graphics:** image, jpeg, graphics
- ❖ **Sci.space:** space, nasa, orbit
- ❖ **Soc.religion.christian:** god, church, sin
- ❖ **Talk.politics.mideast:** israel, armenian, turkish
- ❖ **Talk.religion.misc:** jesus, god, jehovah

*Bekkerman et al,
JMLR, 2003*

特征选择的应用领域



特征选择问题

设有 n 个可用作分类的特征，为了在**不降低（或者尽量不降低）分类精度**的前提下，减小特征空间的维数以减少计算量，需要从中**直接**选出 m 个作为分类特征。这就是特征选择。

目标函数：

- (1) 不降低分类精度
- (2) 特征个数越少越好

从 n 个特征中，选出 m 个特征，一共有

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

种可能的选法。如果对每一种选法进行测试计算其错误率，固然能得到最佳的特征组合，但是计算量太大。因此这种穷举的方法并不实用。

1.1 单变量选择法

单变量选择法就是把n维特征每个分量单独使用时的可分性准则函数值都算出来，然后按准则函数值按从大到小排序，如

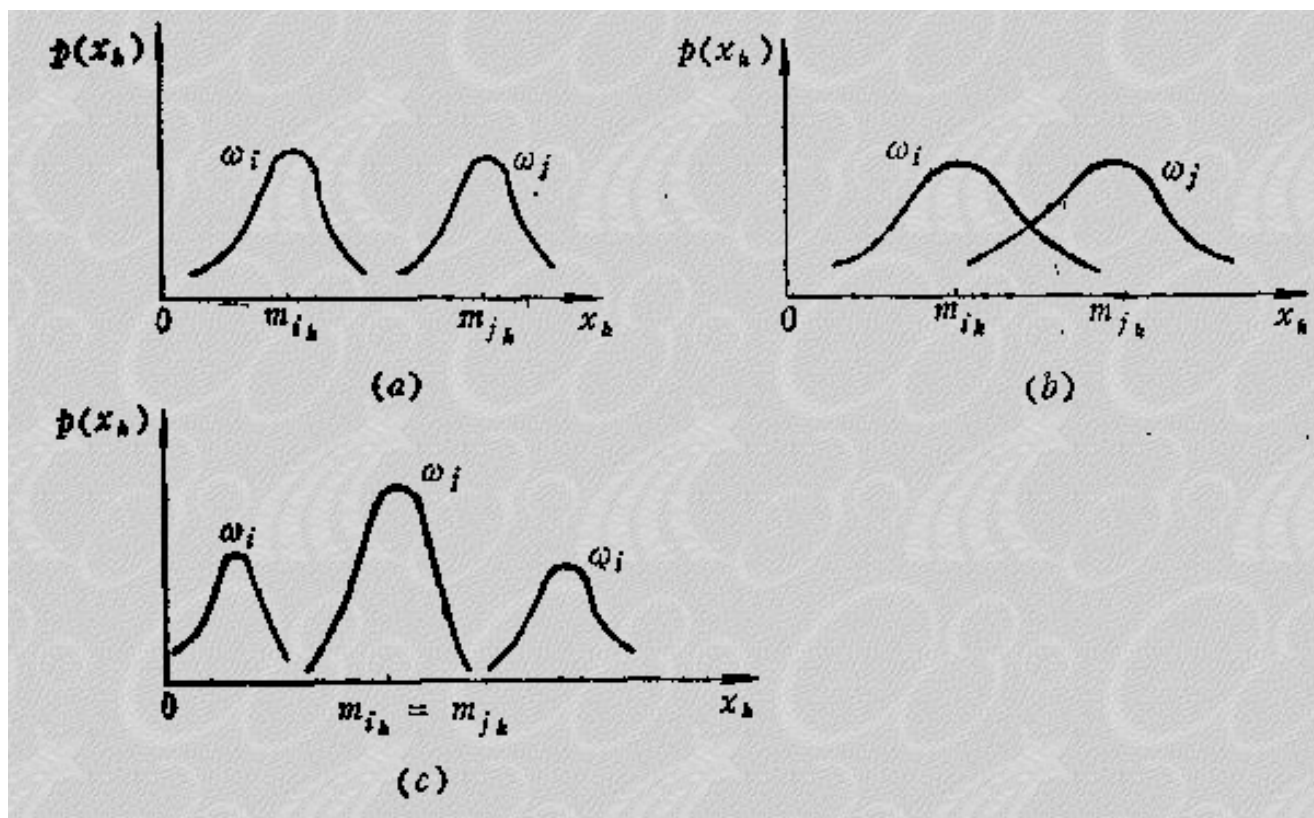
$$G(x_1) > G(x_2) > \cdots > G(x_m) > \cdots > G(x_n)$$

然后，取使 G 较大的前m个特征作为选择结果。

如两类样本 w_i 和 w_j ， $G(x_k)$ 可定义为：

$$G(x_k) = \frac{(m_{ik} - m_{jk})^2}{\sigma_{ik}^2 + \sigma_{jk}^2}, \quad k = 1, 2, \cdots, n$$

该方法简单，但适用范围与模式特征的概率分布有关。
当类概率密度不能用正态分布近似时，不适用。



相关性评价

- 独立性

$$P(X, Y) = P(X) P(Y)$$

- 相关性：互信息

$$\begin{aligned} \text{MI}(X, Y) &= \int P(X, Y) \log \frac{P(X, Y)}{P(X)P(Y)} dX dY \\ &= \text{KL}(P(X, Y) \parallel P(X)P(Y)) \end{aligned}$$

Kullback-Leibler散度

- 熵
$$H(X) \equiv \sum_{x \in \mathcal{A}_X} P(x) \log \frac{1}{P(x)},$$

- 相互熵
$$H(X, Y) = \sum_{xy \in \mathcal{A}_X \mathcal{A}_Y} P(x, y) \log \frac{1}{P(x, y)}.$$

- KL散度

$$D_{\text{KL}}(P||Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}.$$

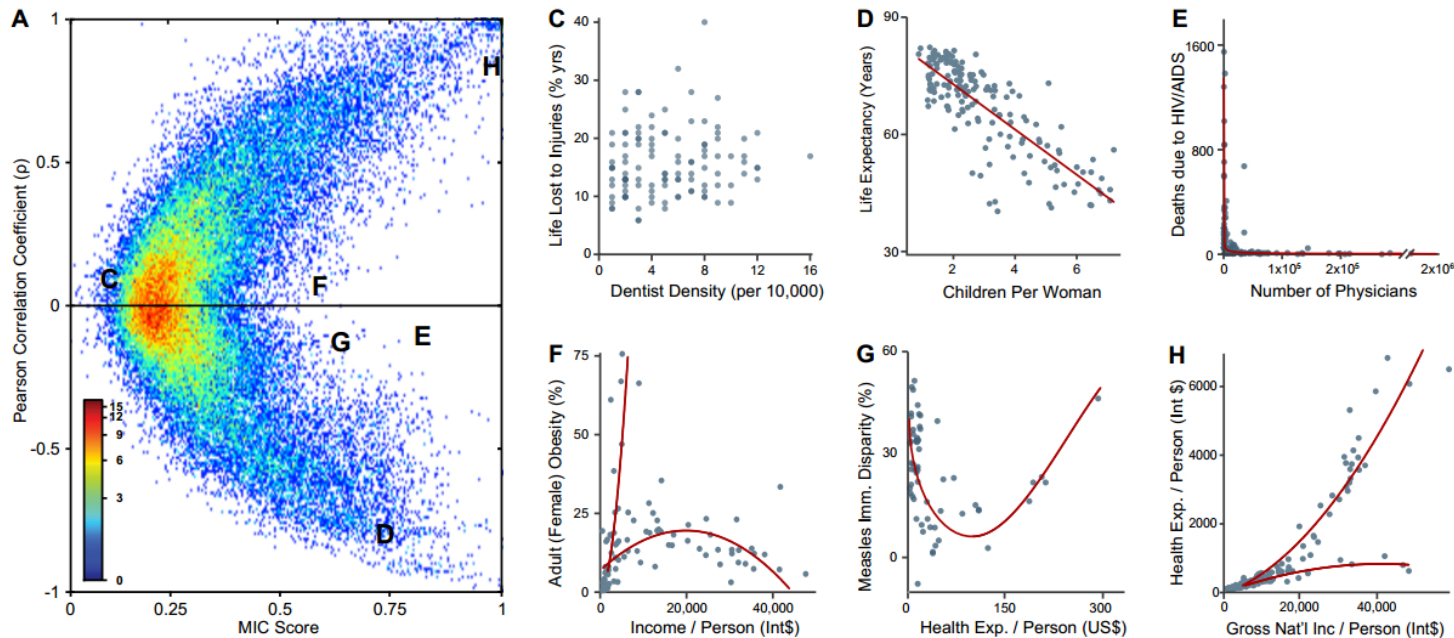
$$I(f_1, f_0) \neq I(f_0, f_1)$$

- Jenson-Shannon熵

$$I_J(f_1, f_0) = I(f_1, f_0) + I(f_0, f_1)$$

Kullback-Leibler散度

- 连续情形

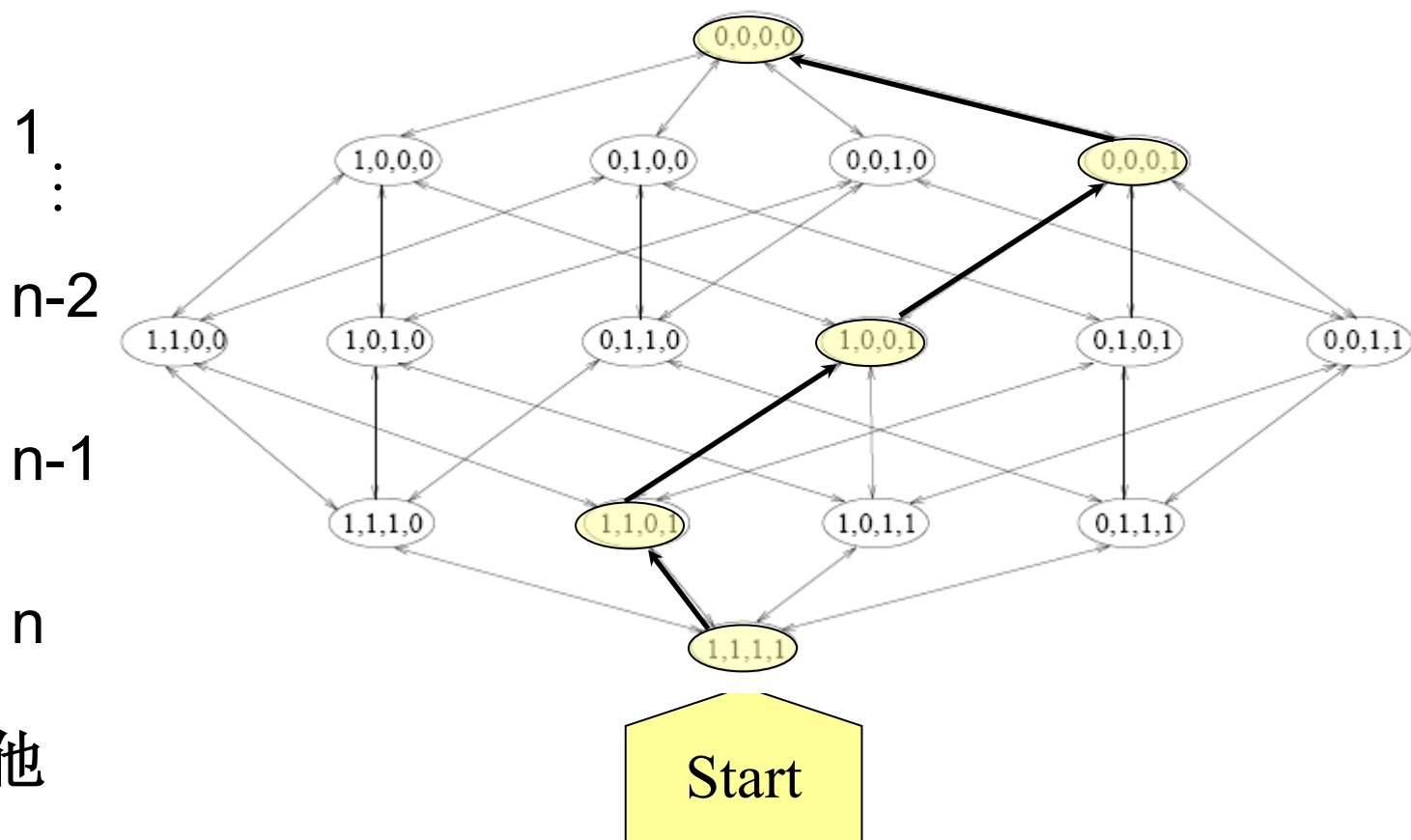


Reshed, D., Y. Reshef, H. Finucane, S. Grossman, G. McVean, P. Turnbaugh, E. Lander, M. Mitzenmacher, and P. Sabeti (2011, December). Detecting novel associations in large data sets. *Science* 334, 1518–1524.

1.2 搜索法

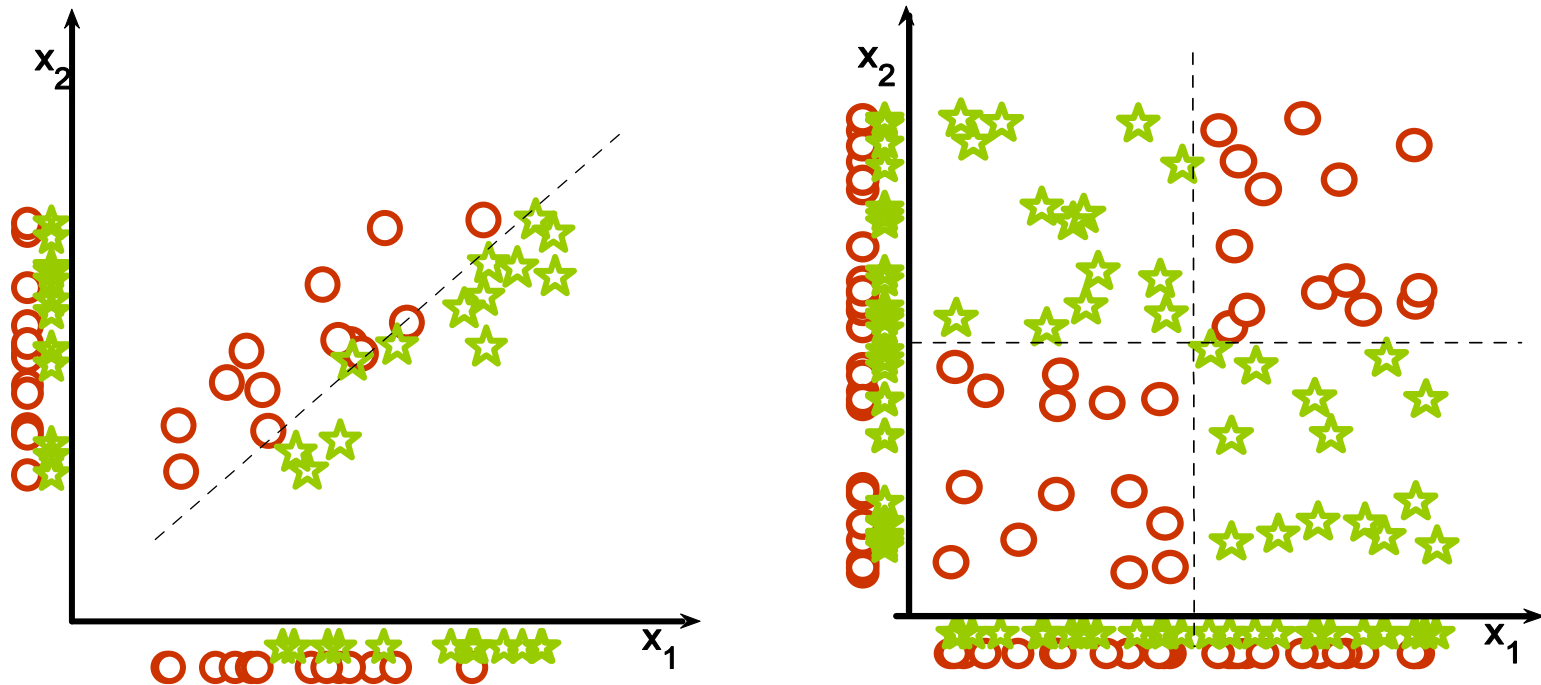
❖ 前向搜索 $(n(n+1))/2$

❖ 后向搜索 $(n(n+1))/2$



❖ 其他

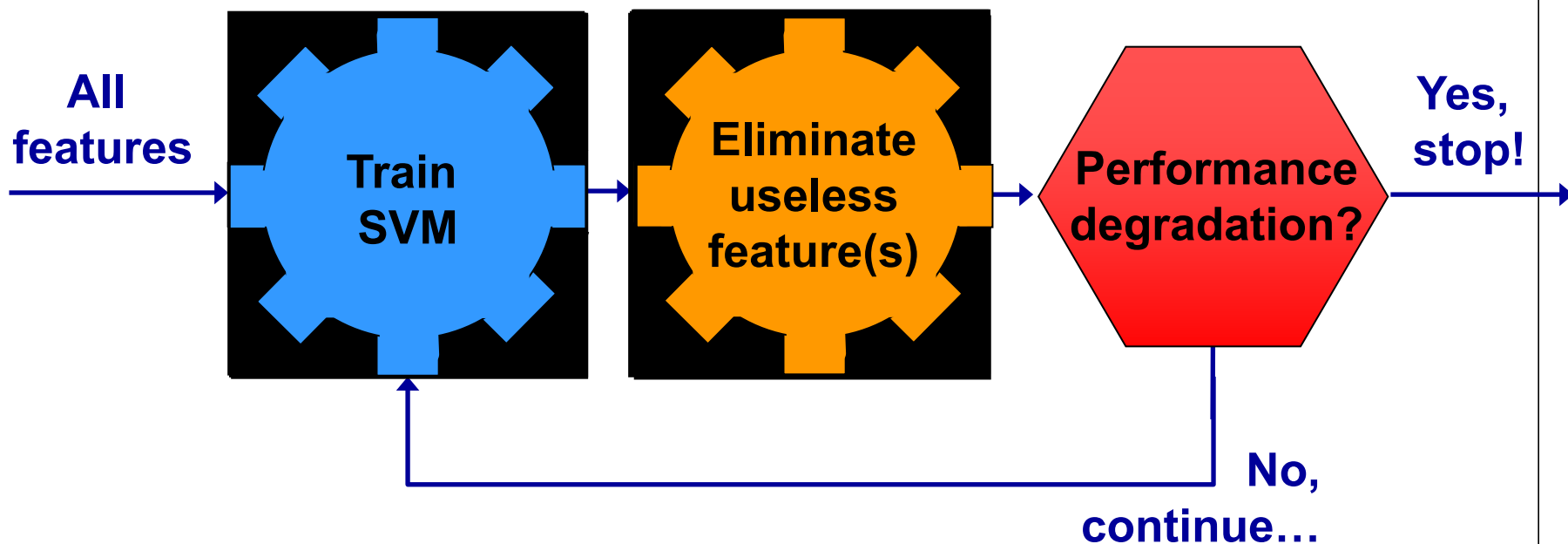
问题与不足



Guyon-Elisseff, JMLR 2004; Springer 2006

1.3 嵌入式方法

- 利用某些分类器、先验分布的性质，自适应地检测并去除冗余特征。
- 如：基于支持向量机

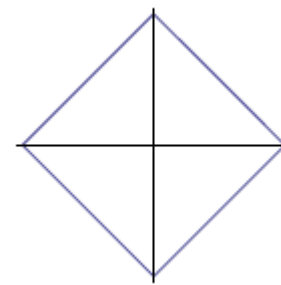


Recursive Feature Elimination (RFE) SVM. *Guyon-Weston, 2000. US patent 7,117,188*

基于稀疏先验分布

- 拉普拉斯分布

$$p_L(w) = \left(\frac{\alpha}{2}\right) \exp(-\alpha|w|)$$



- 目标函数

$$\min_{\mathbf{w}} \sum_{i=1}^N \log(1 + \exp(-\mathbf{w}^T \bar{\mathbf{z}}_i)) + \lambda \|\mathbf{w}\|_1, \text{ subject to } \mathbf{w} \geq 0$$

Krishnapuram, B., Figueiredo, M., Carin, L., & Hartemink, A. (2005) “Sparse Multinomial Logistic Regression: Fast Algorithms and Generalization Bounds.” IEEE Trans. Pattern Analysis and Machine Intelligence, 27, 2005. pp. 957–968.

嵌入式方法结果

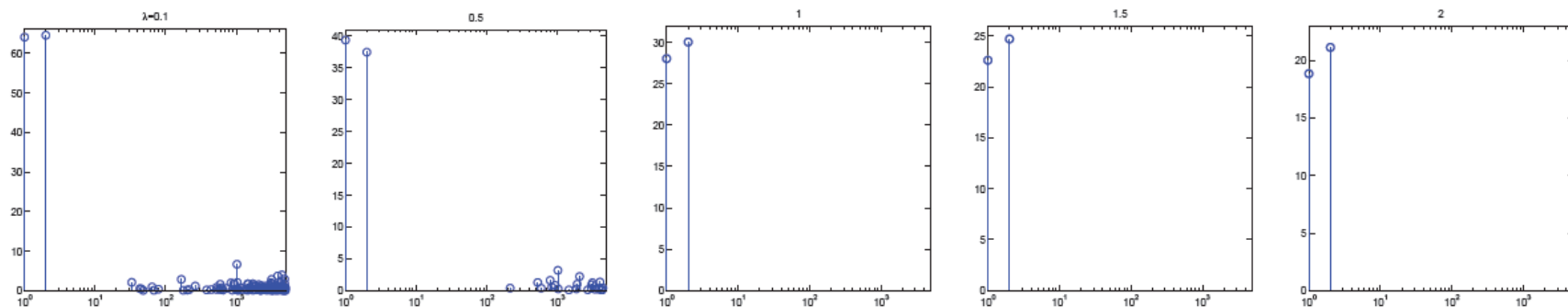


Fig. 6. Feature weights learned on the spiral data with 5000 irrelevant features, for a fixed kernel width $\sigma = 2$ and different regularization parameters $\lambda \in \{0.1, 0.5, 1, 1.5, 2\}$.

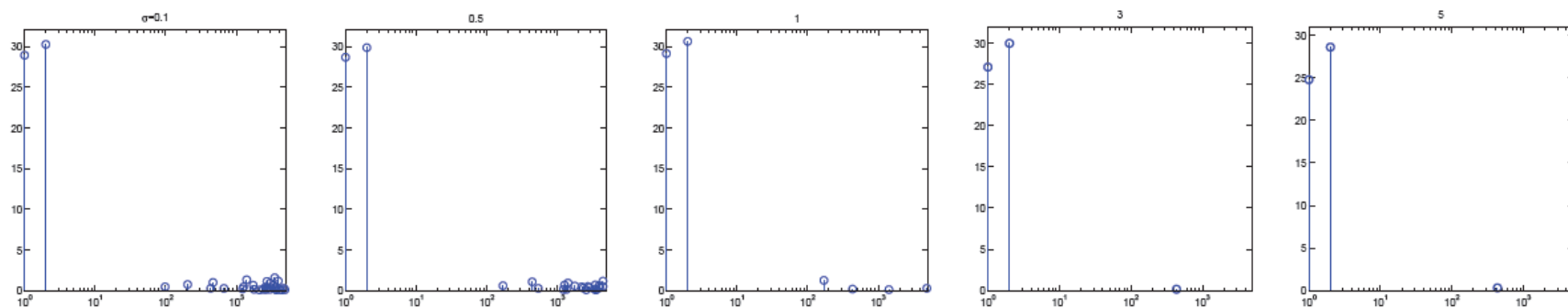


Fig. 7. Feature weights learned on the spiral data with 5000 irrelevant features, for a fixed regularization parameter $\lambda = 1$ and different kernel widths $\sigma \in \{0.1, 0.5, 1, 3, 5\}$.

2. 特征提取

- 预备知识
 - 矩阵分析基础
- 线性方法
 - 主成分分析
 - 线性判别分析
- 非线性方法
 - 流形学习
 - 核特征提取方法

2.1 矩阵分析基础

- 矩阵 A
 - 基本操作：加法、乘法
 - 其它重要操作：转置、 $\text{inv}(A)$ 、 rank (秩)、 $\text{det}(A)$ 、迹 (trace)
- 特征值和特征向量
 - (λ, \mathbf{v}) 是一个特征值-特征向量对，如果满足

$$Ax = \lambda x$$

- λ ：特征值
- \mathbf{v} ：特征向量

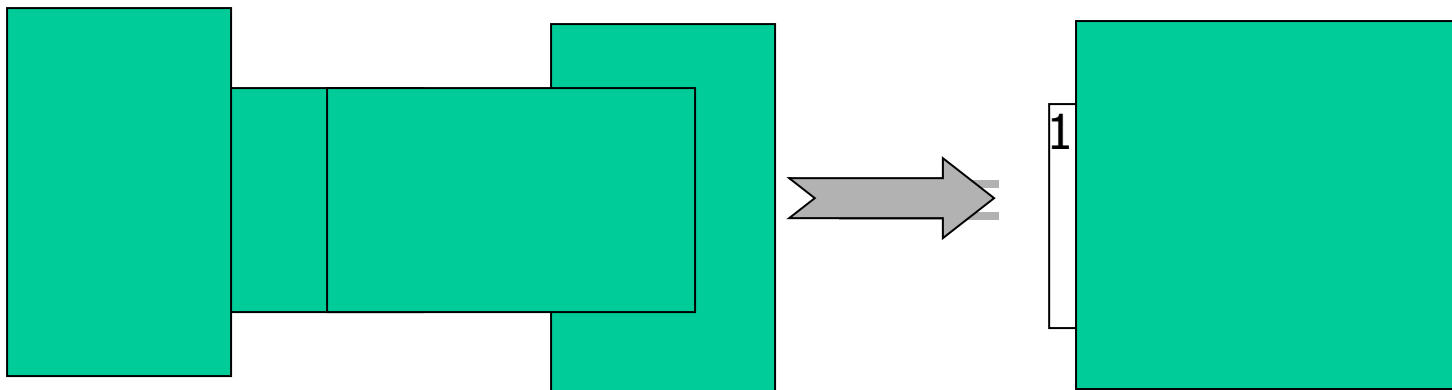
正交矩阵

❖ 矩阵 $U \in \mathfrak{R}^{m \times m}$ 是正交矩阵，当且仅当

$$U U^T = I_m \Rightarrow U^{-1} = U^T$$

❖ 矩阵 $V \in \mathfrak{R}^{m \times n}$ ($m > n$) 是列正交矩阵，当且仅当

$$V^T V = I$$



矩阵范数和迹 (Trace)

矩阵范数(norm):

2-norm: $\|A\|_2 = \sqrt{\lambda_{\max}(AA^T)}$ 的最大特征值的平方根.

F-norm: $\|A\|_F = \sqrt{\sum_{i,j} A_{ij}^2}$.

1-norm: $\|A\|_1 = \sum_{i,j} |A_{ij}|$.

$\text{trace}(A) = \sum_{i=1}^m A_{ii}$, A 为 $m \times m$ 的正方阵.

$\|A\|_F^2 = \text{trace}(AA^T) = \text{trace}(A^T A)$, $\text{trace}(AB) = \text{trace}(BA)$.

$\|QA\|_F = \|A\|_F$, 如果 Q 为列正交矩阵.

对称矩阵、正定矩阵和QR分解

如果 $A = A^T$, 称 A 是对称的.

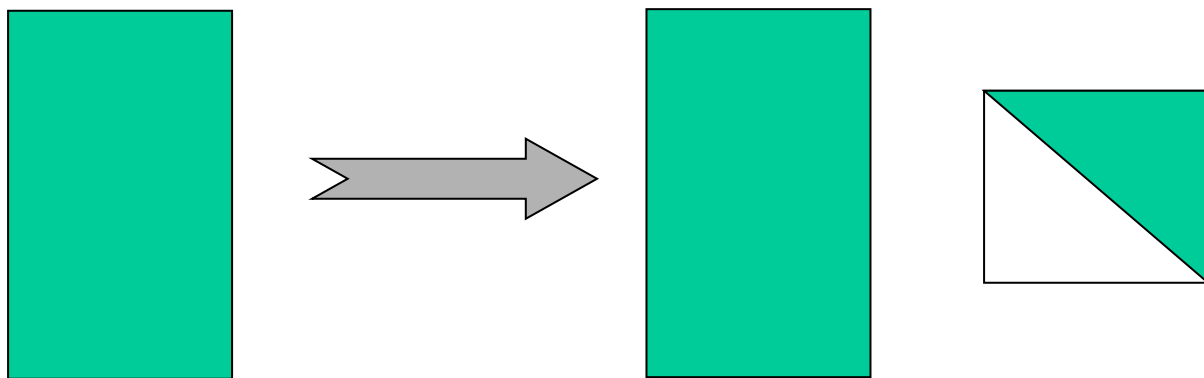
如果 对任意 $x \in \mathbb{R}^m$, $x^T A x \geq 0$, $A \in \mathbb{R}^{m \times m}$ 是对称半正定矩阵.

如果 对任意 非零 $x \in \mathbb{R}^m$, $x^T A x > 0$, $A \in \mathbb{R}^{m \times m}$ 是对称正定矩阵.

QR分解: $A = QR$, 其中 $Q \in \mathbb{R}^{m \times n}$ 的列向量相互正交,

$R \in \mathbb{R}^{n \times n}$ 为上三角非奇异矩阵.

(假设 $m > n$ 并且 A 的列向量线性 独立.)

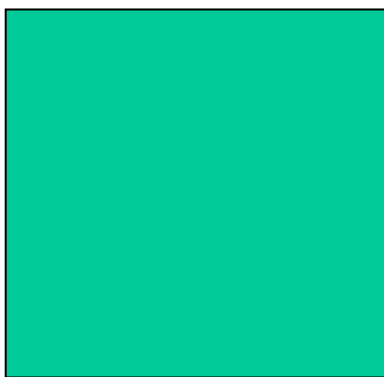
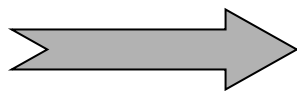
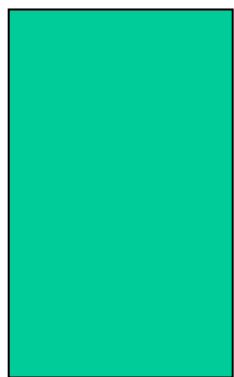


奇异值分解 (SVD)

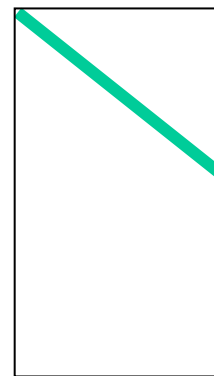
奇异值分解 (SVD): $A = U\Sigma V^T$, 其中 $A \in \mathbb{R}^{m \times n}$, $U \in \mathbb{R}^{m \times m}$, $V \in \mathbb{R}^{n \times n}$ 为正交矩阵, $\Sigma = \text{diag}(\sigma_1, \dots, \sigma_r)$ 是对角矩阵并有 $\sigma_1 \geq \dots \geq \sigma_r$, $r = \min(m, n)$.

$AA^T = U\Sigma\Sigma^T U^T$: U 是矩阵 AA^T 的特征向量集合.

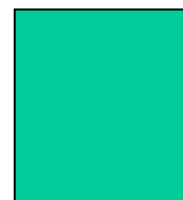
$A^T A = V\Sigma^T \Sigma V^T$: V 是矩阵 $A^T A$ 的特征向量集合.



正交



对角



正交

SVD性质

THEOREM 2.1. *Let the SVD of A be given by Equation (1) and*

$$\sigma_1 \geq \sigma_2 \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0$$

and let $R(A)$ and $N(A)$ denote the range and null space of A , respectively. Then,

- 1. rank property: $\text{rank}(A) = r$, $N(A) \equiv \text{span}\{v_{r+1}, \dots, v_n\}$, and $R(A) \equiv \text{span}\{u_1, \dots, u_r\}$, where $U = [u_1 u_2 \cdots u_m]$ and $V = [v_1 v_2 \cdots v_n]$.*
- 2. dyadic decomposition: $A = \sum_{i=1}^r u_i \cdot \sigma_i \cdot v_i^T$.*
- 3. norms: $\|A\|_F^2 = \sigma_1^2 + \cdots + \sigma_r^2$, and $\|A\|_2 = \sigma_1$.*

SVD性质

THEOREM 2.2. [Eckart and Young] *Let the SVD of A be given by Equation (1) with $r = \text{rank}(A) \leq p = \min(m, n)$ and define*

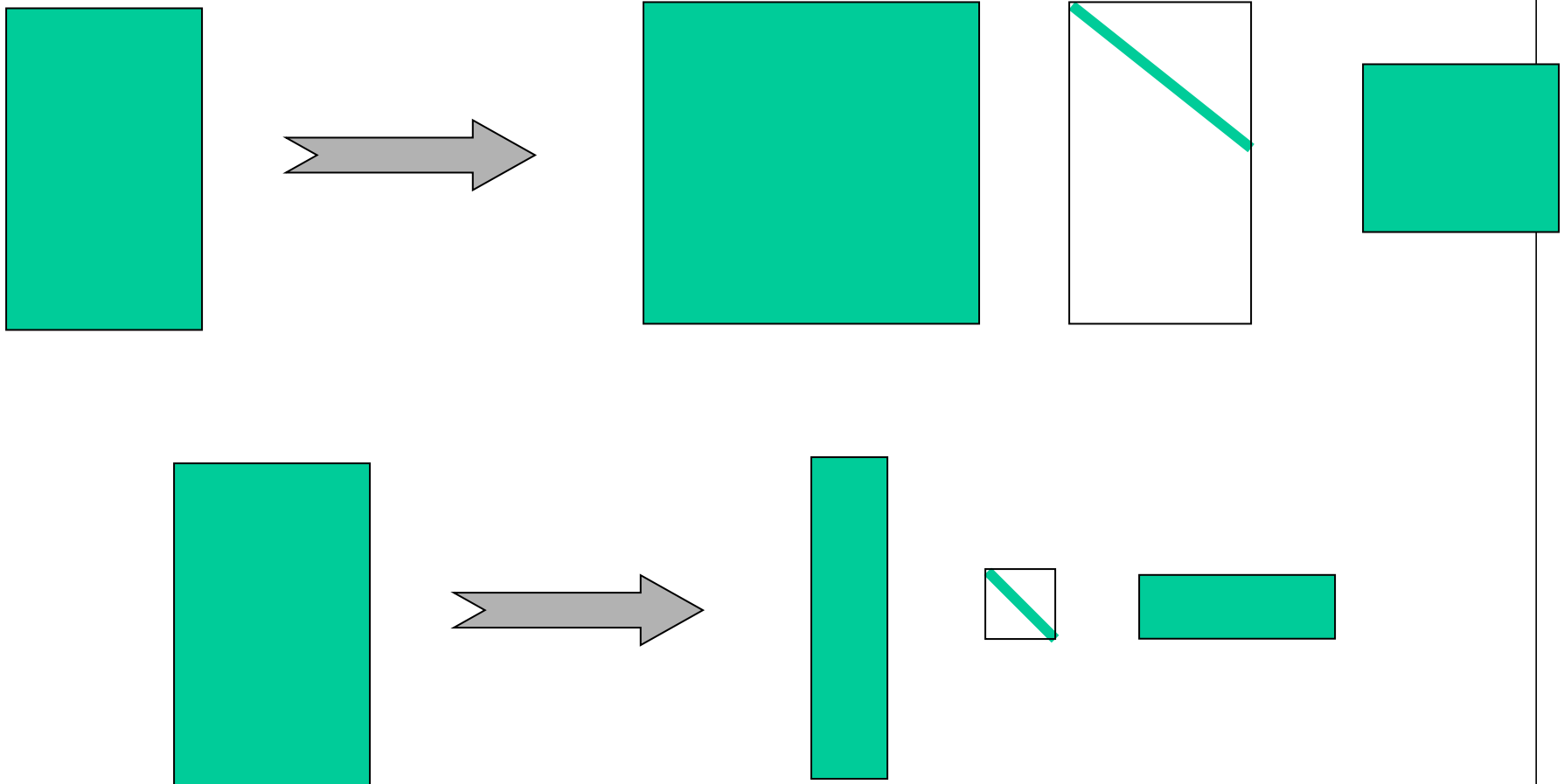
$$(2) \quad A_k = \sum_{i=1}^k u_i \cdot \sigma_i \cdot v_i^T ,$$

then

$$\min_{\text{rank}(B)=k} \|A - B\|_F^2 = \|A - A_k\|_F^2 = \sigma_{k+1}^2 + \cdots + \sigma_p^2.$$

- A_k 是矩阵Frobenius范数下的k-秩最优逼近

低秩逼近



问题的提出

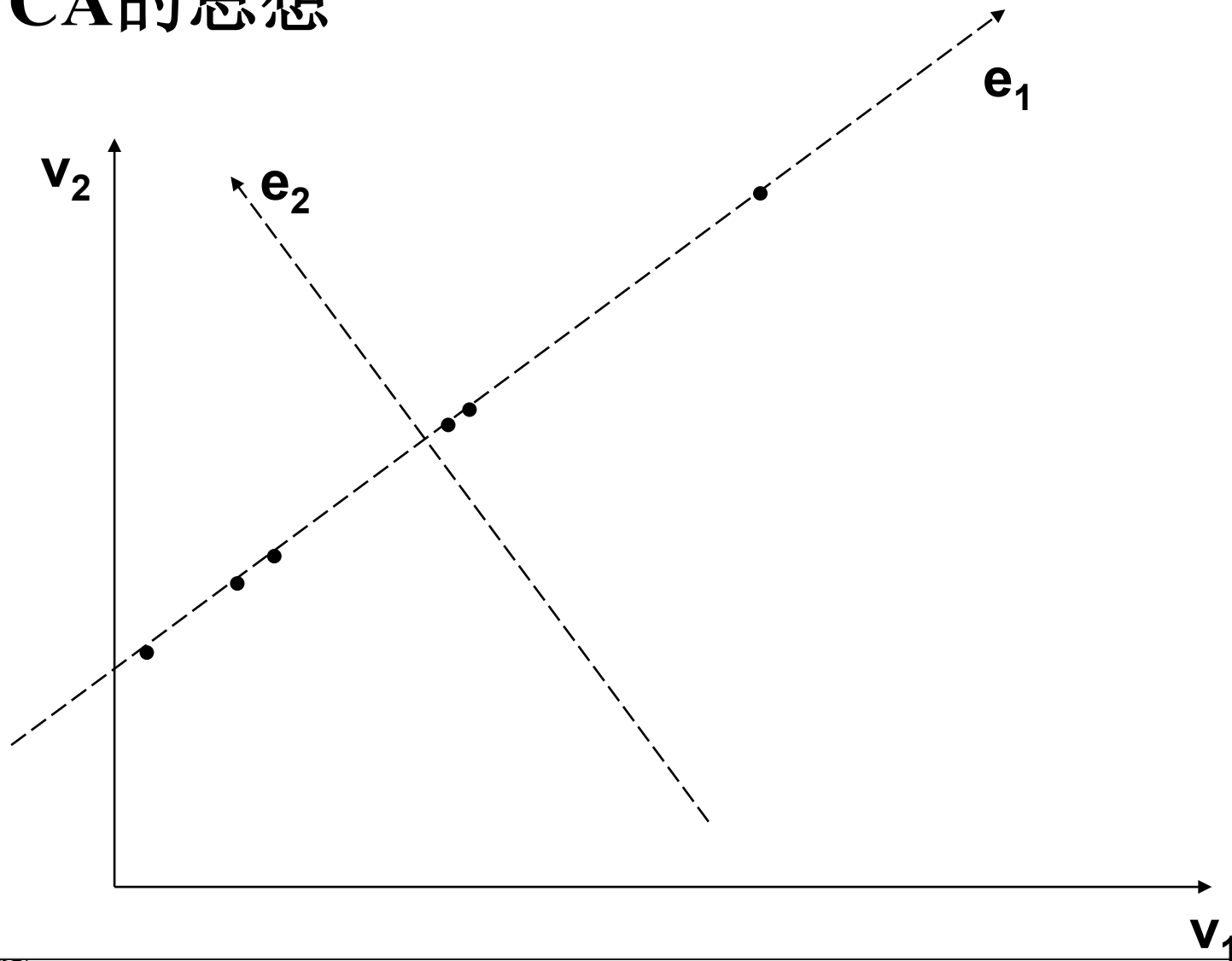
- 一般来说，在建立识别系统时，抽取的原始特征往往比较多，特征的维数比较大，这会给识别器的训练带来很大的困难，因此希望能够采用某种方法降低特征的维数。这些方法可以称作**成分分析**的方法。
- 成分分析方法主要包括：
 1. **主成分分析**；
 2. 线性判别分析；
 3. 独立成分分析；

2.1 主成分分析

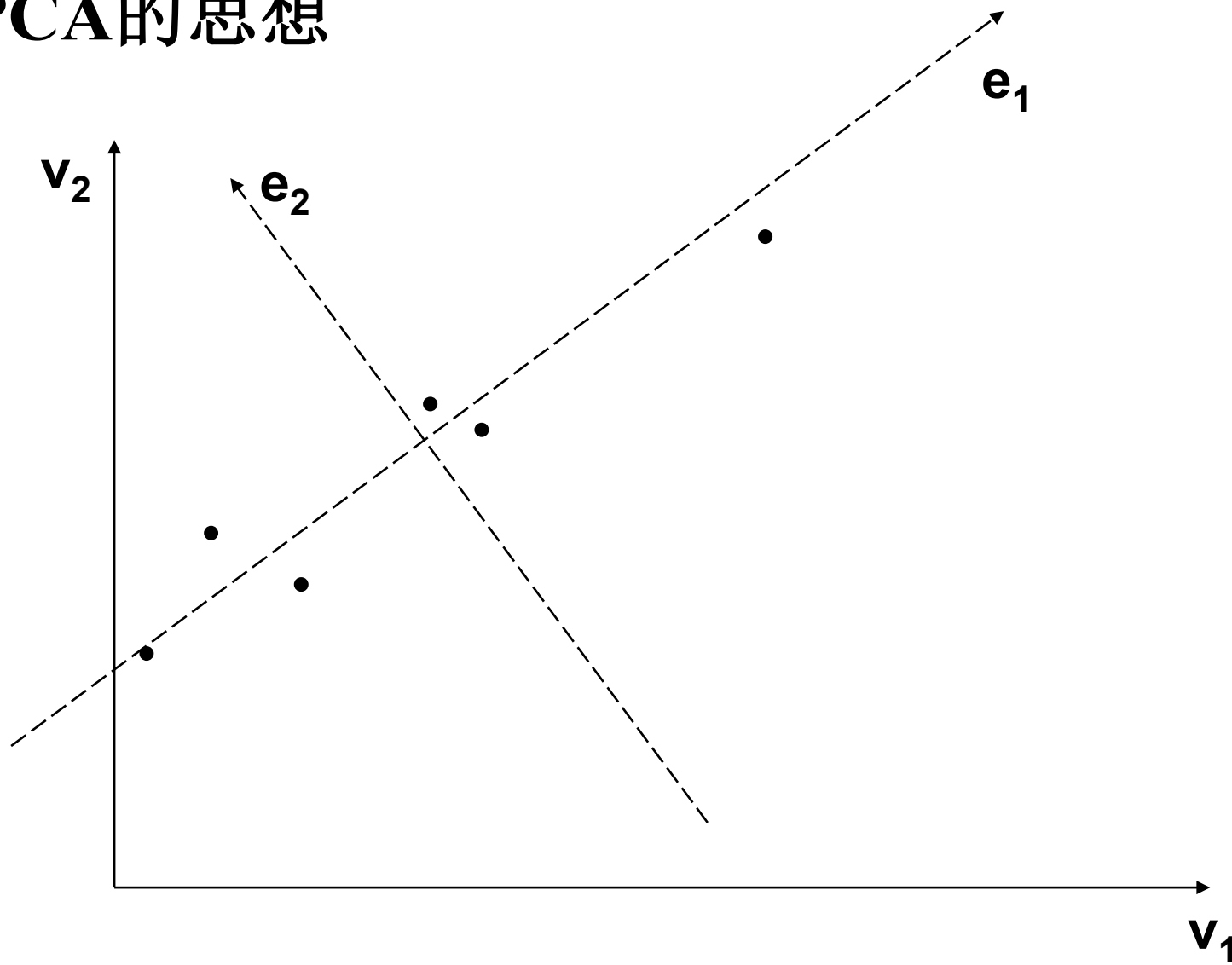
(PCA, Principal Component Analysis)

- PCA是一种最常用的**线性**成分分析方法;
- PCA的主要思想是寻找到数据的主轴方向, 由主轴构成一个新的坐标系 (维数可以比原维数低), 然后数据由原坐标系向新的坐标系投影。
- PCA的**其它名称**: 离散K-L变换, Hotelling变换;

PCA的思想



PCA的思想



目标（损失）

- 输入：训练样本集 $(\mathbf{x}_1, \dots, \mathbf{x}_N)$ ，其维数为 n
- 输出：样本的均值 \mathbf{m} 和 d ($< n$) 个主成分方向 $(\mathbf{u}_1, \dots, \mathbf{u}_d)$ 。
- 损失函数

$$\min_{\mathbf{m}, \mathbf{u}_1, \dots, \mathbf{u}_d} \frac{1}{N} \sum_{i=1}^N \left\| \mathbf{x}_i - \mathbf{m} - \sum_{j=1}^d \mathbf{u}_j \mathbf{u}_j^T (\mathbf{x}_i - \mathbf{m}) \right\|^2$$
$$\text{s.t. } \mathbf{u}_i^T \mathbf{u}_j = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}$$

PCA算法

1. 利用训练样本集合计算样本的均值 \mathbf{m} 和协方差矩阵（散度矩阵） \mathbf{S} ;
2. 计算 \mathbf{S} 的特征值，并由大到小排序;
3. 选择前 d' 个特征值对应的特征向量作成变换矩阵 $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{d'}]$;
4. 训练和识别时，每一个输入的 d 维特征向量 \mathbf{x} 可以转换为 d' 维的新特征向量 \mathbf{y} :

$$\mathbf{y} = \mathbf{E}^t(\mathbf{x} - \mathbf{m}).$$

Principal Component Analysis (PCA)

- Methodology

- Suppose x_1, x_2, \dots, x_M are $N \times 1$ vectors

Step 1: $\bar{x} = \frac{1}{M} \sum_{i=1}^M x_i$

Step 2: subtract the mean: $\Phi_i = x_i - \bar{x}$

Step 3: form the matrix $A = [\Phi_1 \ \Phi_2 \ \cdots \ \Phi_M]$ ($N \times M$ matrix), then compute:

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = A A^T \frac{1}{M}$$

(sample **covariance** matrix, $N \times N$, characterizes the *scatter* of the data)

Step 4: compute the eigenvalues of C : $\lambda_1 > \lambda_2 > \cdots > \lambda_N$

Step 5: compute the eigenvectors of C : u_1, u_2, \dots, u_N

Principal Component Analysis (PCA)

- Methodology – cont.

- Since C is symmetric, u_1, u_2, \dots, u_N form a basis, (i.e., any vector x or actually $(x - \bar{x})$, can be written as a linear combination of the eigenvectors):

$$x - \bar{x} = b_1 u_1 + b_2 u_2 + \dots + b_N u_N = \sum_{i=1}^N b_i u_i$$

Step 6: (dimensionality reduction step) keep only the terms corresponding to the K largest eigenvalues:

$$\hat{x} - \bar{x} = \sum_{i=1}^K b_i u_i \text{ where } K \ll N$$

- The representation of $\hat{x} - \bar{x}$ into the basis u_1, u_2, \dots, u_K is thus

$$\begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_K \end{bmatrix}$$

Principal Component Analysis (PCA)

- Linear transformation implied by PCA
 - The linear transformation $R^N \rightarrow R^K$ that performs the dimensionality reduction is:

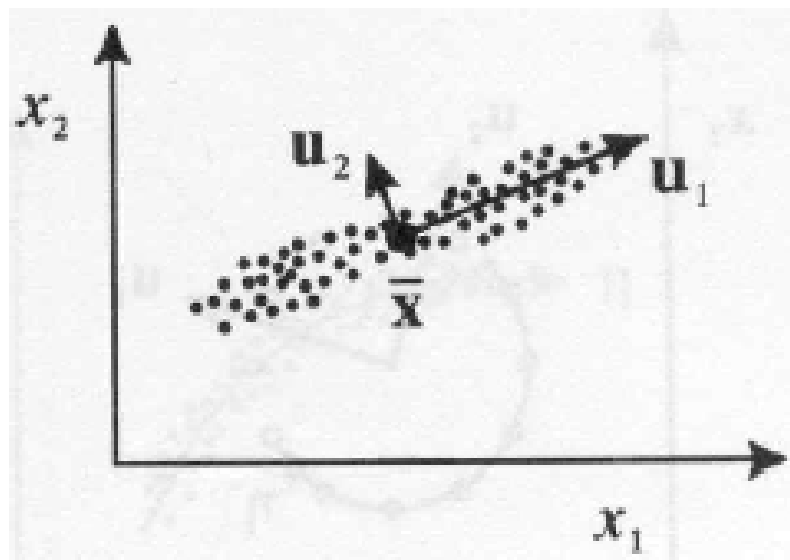
$$\begin{bmatrix} b_1 \\ b_2 \\ \dots \\ b_K \end{bmatrix} = \begin{bmatrix} u_1^T \\ u_2^T \\ \dots \\ u_K^T \end{bmatrix} (x - \bar{x}) = U^T (x - \bar{x})$$

(i.e., simply computing coefficients of linear expansion)

The above expression assumes that u_i has unit length (i.e., normalized)

Principal Component Analysis (PCA)

- Geometric interpretation
 - PCA projects the data along the directions where the data varies the most.
 - These directions are determined by the eigenvectors of the covariance matrix corresponding to the largest eigenvalues.
 - The magnitude of the eigenvalues corresponds to the variance of the data along the eigenvector directions.



Principal Component Analysis (PCA)

- How to choose the principal components?
 - To choose K , use the following criterion:

$$\frac{\sum_{i=1}^K \lambda_i}{\sum_{i=1}^N \lambda_i} > \textit{Threshold} \quad (\text{e.g., } 0.9 \text{ or } 0.95)$$

- In this case, we say that we “preserve” 90% or 95% of the information in our data.
- If $K=N$, then we “preserve” 100% of the information in our data.

Principal Component Analysis (PCA)

- Error due to dimensionality reduction
 - The original vector x can be reconstructed using its principal components:

$$\hat{x} - \bar{x} = \sum_{i=1}^K b_i u_i \text{ or } \hat{x} = \sum_{i=1}^K b_i u_i + \bar{x}$$

- It can be shown that the low-dimensional basis based on principal components minimizes the reconstruction error:

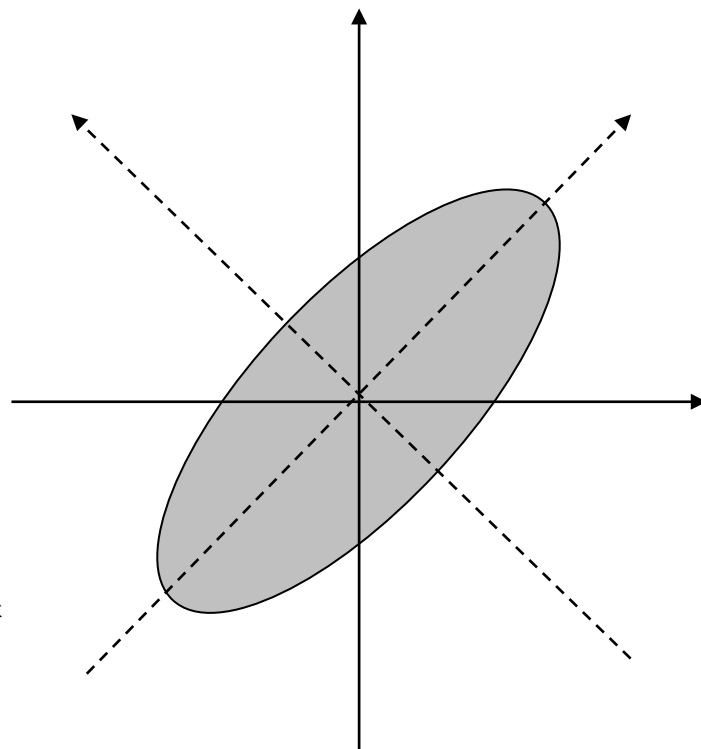
$$e = ||x - \hat{x}||$$

- It can be shown that the error is equal to:

$$e = 1/2 \sum_{i=K+1}^N \lambda_i$$

PCA的讨论

- 由于 S 是实对称阵，因此特征向量是**正交**的；
- 将数据向新的坐标轴投影之后，特征之间是**不相关**的；
- 特征值描述了变换后各维特征的重要性，特征值为0的各维特征为冗余特征，可以去掉。



PCA性质

- 最大方差

$$\text{var}[z_1] = E((z_1 - \bar{z}_1)^2) = \frac{1}{n} \sum_{i=1}^n (a_1^T x_i - a_1^T \bar{x})^2$$

$$= \frac{1}{n} \sum_{i=1}^n a_1^T (x_i - \bar{x})(x_i - \bar{x})^T a_1 = a_1^T S a_1$$

- 最优重建

$$\min_{G \in \mathbb{R}^{p \times d}} \|X - G(G^T X)\|_F^2 \text{ subject to } G^T G = I_d$$

- 去共生（相关）

$$\text{cov}[z_2, z_1] = 0$$

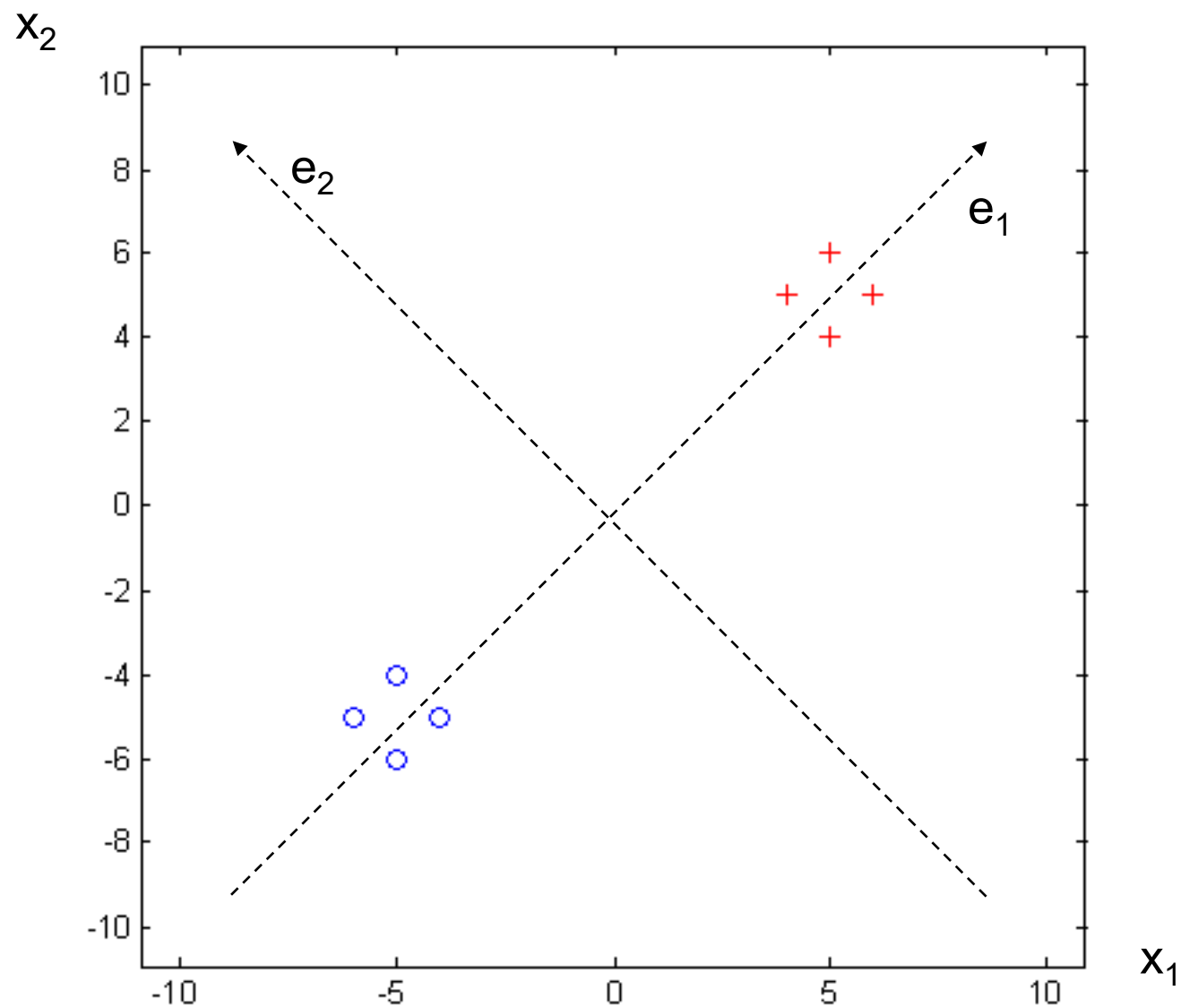
例8.1

- 有两类问题的训练样本：

$$\omega_1 : (-5, -4)^t, (-4, -5)^t, (-5, -6)^t, (-6, -5)^t$$

$$\omega_2 : (5, 4)^t, (4, 5)^t, (5, 6)^t, (6, 5)^t$$

将特征由2维压缩为1维。



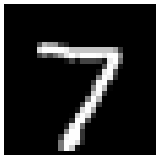
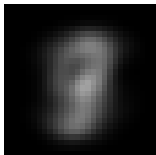
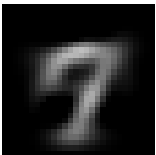
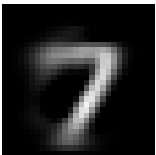
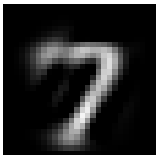




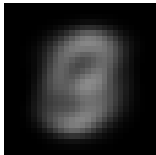
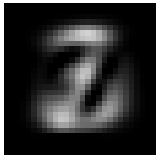
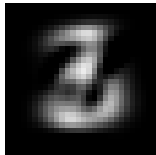
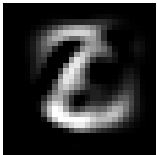



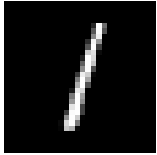
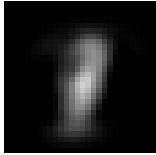
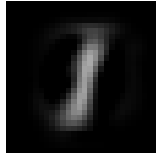
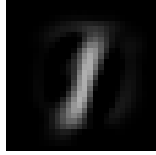
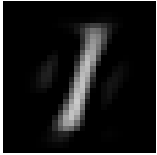
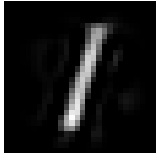
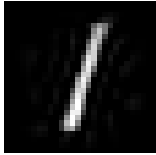
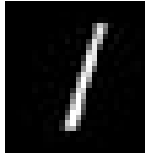
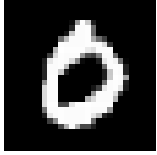
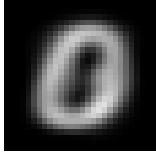
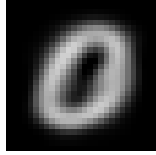
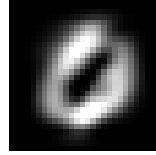
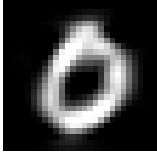
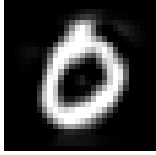
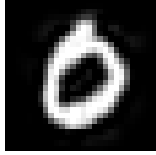
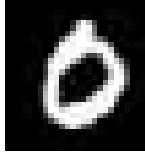


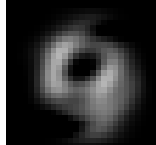
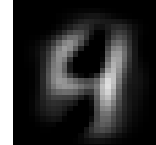




人脸识别举例



特征人脸 (Eigenfaces)

 \mathbf{e}_1 \mathbf{e}_2 \mathbf{e}_3 \mathbf{e}_4 \mathbf{e}_5 \mathbf{e}_6 \mathbf{e}_7 \mathbf{e}_8 

PCA重构

原图像	$d'=1$	5	10	20	50	100	200
							
							
							
							
							

PCA性质

- 最大方差 $\text{var}[z_1] = E((z_1 - \bar{z}_1)^2) = \frac{1}{n} \sum_{i=1}^n (a_1^T x_i - a_1^T \bar{x})^2$

$$= \frac{1}{n} \sum_{i=1}^n a_1^T (x_i - \bar{x})(x_i - \bar{x})^T a_1 = a_1^T S a_1$$

- 最优重建

$$\min_{G \in \mathbb{R}^{p \times d}} \|X - G(G^T X)\|_F^2 \text{ subject to } G^T G = I_d$$

- 去共生

$$\text{cov}[z_2, z_1] = 0$$

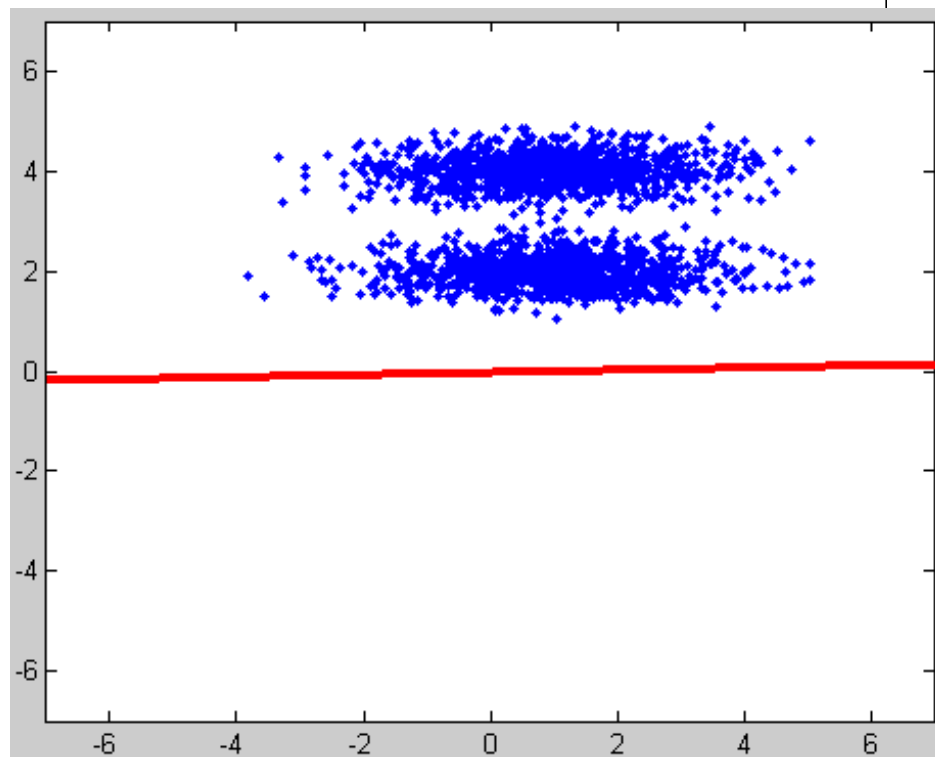
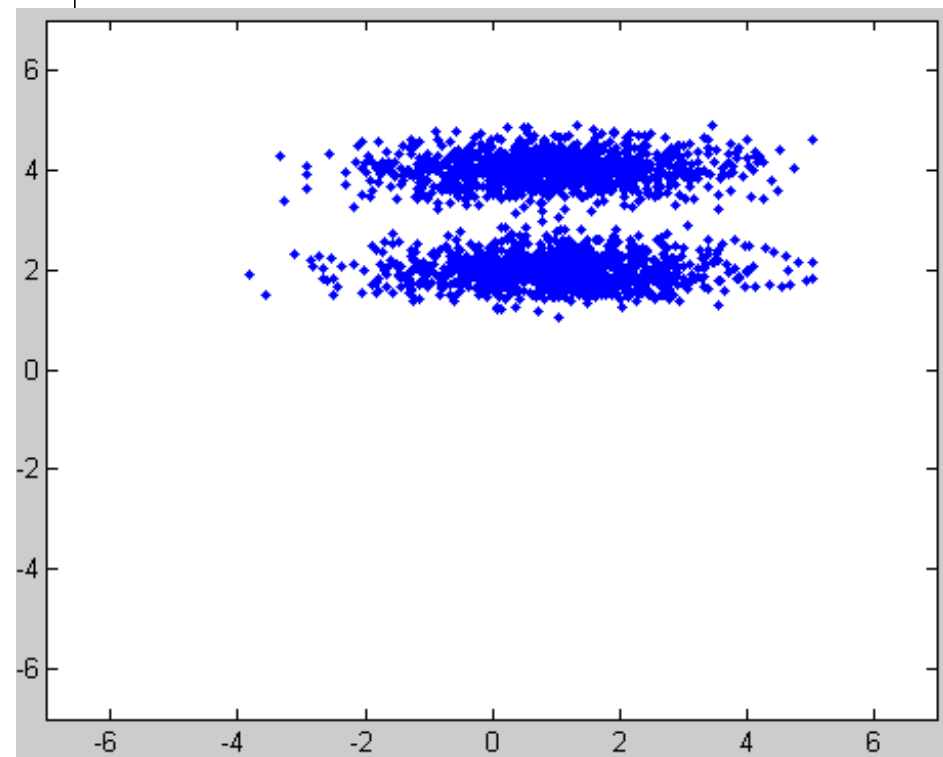
PCA -> LDA

- PCA将所有的样本作为一个整体对待，寻找一个均方误差最小意义下的最优线性映射，而没有考虑样本的类别属性，它所忽略的投影方向有可能恰恰包含了重要的可分性信息；
- LDA则是在可分性最大意义下的最优线性映射，充分保留了样本的类别可分性信息；
- LDA还被称为FDA(Fisher Discriminant Analysis) 。

线性判别分析(LDA)

- PCA

$$Y = w^T X$$

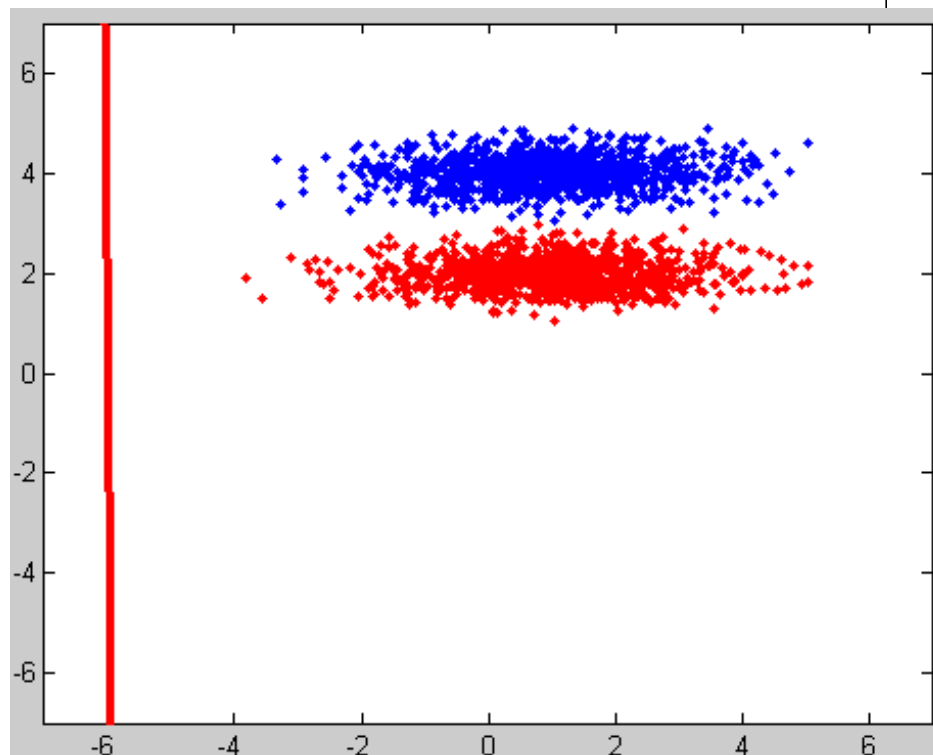
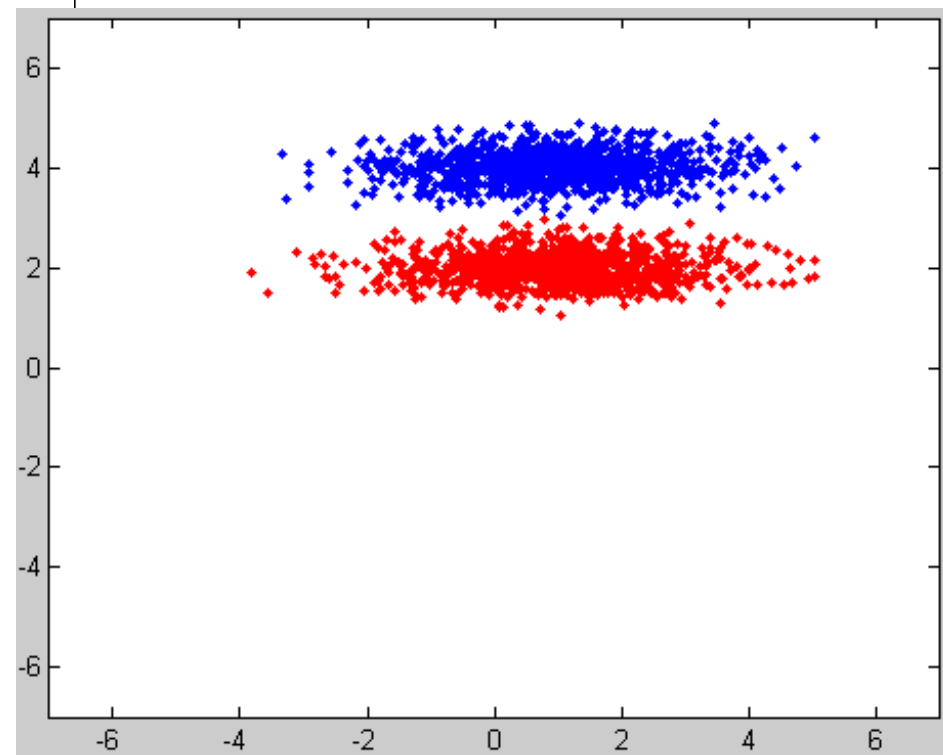


寻求变换 w , 使得 $w^T x$ 的分布最大限度的分散 (**最大方差原则**)

Linear Discriminant Analysis (LDA)

- LDA

$$Y_1 = w^T X_1 \quad Y_2 = w^T X_2$$



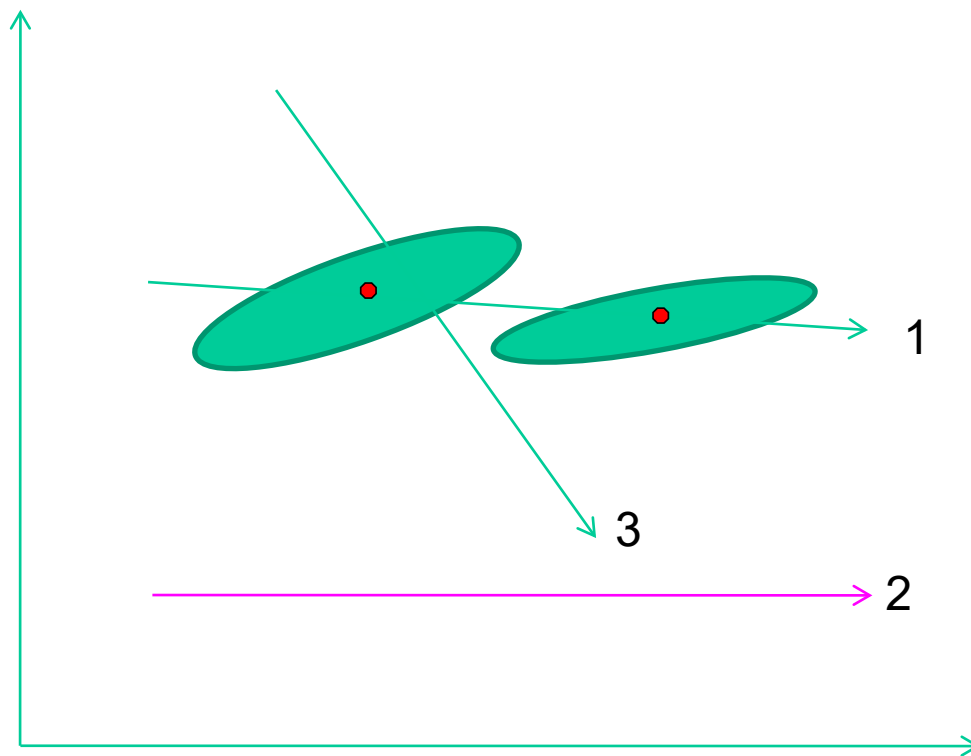
寻求变换 w , 最大限度地拉大 $w^T X_1$ 和 $w^T X_2$ 的距离, 同时使得类样本的类内分布的分散程度最小 (最大分离原则)

线性判别分析(LDA)

- 基本假设：
 - 各类的类内分布相同
 - 满足假设时理论最优
- 维数约简
- 尽可能的保持原始数据中的类别判别信息.
- 寻求使各类样本之间分离程度最大的投影方向.
- 同时考虑了类内散度分布和类间散度分布情况.
- 广泛应用
 - 人脸识别
 - 生物信息学

线性判别分析(LDA)

- 只考虑均值和类内方差的情况



线性判别分析(LDA)

- 两类情形: ω_1, ω_2
- 定义类内散度矩阵 (S_w) 和类间散度矩阵 (S_b) 如下

$$Y = w^T X$$

$$\mu_i = \frac{1}{N_i} \sum_{x \in \omega_i} x \quad \tilde{\mu}_i = \frac{1}{N_i} \sum_{y \in \omega_i} y$$

$$S_i = \sum_{x \in \omega_i} (x - \mu_i)(x - \mu_i)^T$$

$$S_w = S_1 + S_2$$

线性判别分析(LDA)

$$\tilde{S}_i^2 = \sum_{y \in \omega_i} (y - \tilde{\mu}_i)^2 = \sum_{x \in \omega_i} (w^T x - w^T \mu_i)^2 = \sum_{x \in \omega_i} w^T (x - \mu_i)(x - \mu_i)^T w = w^T S_i w$$

$$\tilde{S}_1^2 + \tilde{S}_2^2 = w^T S_W w$$

$$(\tilde{\mu}_1 - \tilde{\mu}_2)^2 = (w^T \mu_1 - w^T \mu_2)^2 = w^T (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T w = w^T S_B w$$

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

Maximize S_B

Minimize S_W

线性判别分析(LDA)

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \quad \text{如何求解?}$$

$$\text{minimize} \quad -\frac{1}{2} w^T S_B w \quad \text{s.t.} \quad w^T S_W w = 1$$

$$\Lambda(w, \lambda) = -\frac{1}{2} w^T S_B w + \frac{1}{2} \lambda (w^T S_W w - 1)$$

$$\frac{\partial \Lambda}{\partial w} = -S_B w + \lambda S_W w = 0$$

$$S_B w = \lambda S_W w \quad \text{广义特征值问题}$$

$$S_W^{-1} S_B w = \lambda w \quad \text{特征值问题}$$

LDA算法

1. 利用训练样本集合计算类内散度矩阵 S_w 和类间散度矩阵 S_B ;
2. 计算 $S_w^{-1}S_B$ 的特征值;
3. 选择非0的 $c-1$ 个特征值对应的特征向量作成
一个变换矩阵 $W=[w_1, w_2, \dots, w_{c-1}]$;
4. 训练和识别时, 每一个输入的 d 维特征向量 x
可以转换为 $c-1$ 维的新特征向量 y :

$$y = W^t x。$$

例

• 数据

- $X_1 = \{(4,1), (2,4), (2,3), (3,6), (4,4)\}$
- $X_2 = \{(9,10), (6,8), (9,5), (8,7), (10,8)\}$

• 类统计数据

$$\mu_1 = [3.0 \quad 3.6], \quad \mu_2 = [8.4 \quad 7.6]$$

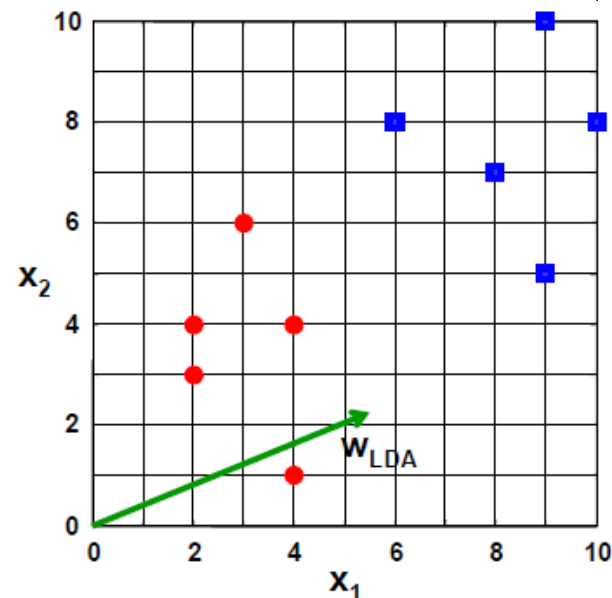
$$S_1 = \begin{bmatrix} .8 & -.4 \\ -.4 & 2.6 \end{bmatrix}, \quad S_2 = \begin{bmatrix} 1.84 & -.04 \\ -.04 & 2.64 \end{bmatrix}$$

• 类内和类间散度

$$S_B = \begin{bmatrix} 29.16 & 21.60 \\ 21.60 & 16.00 \end{bmatrix}, \quad S_W = \begin{bmatrix} 2.64 & -.44 \\ -.44 & 5.28 \end{bmatrix}$$

• 求解特征值问题

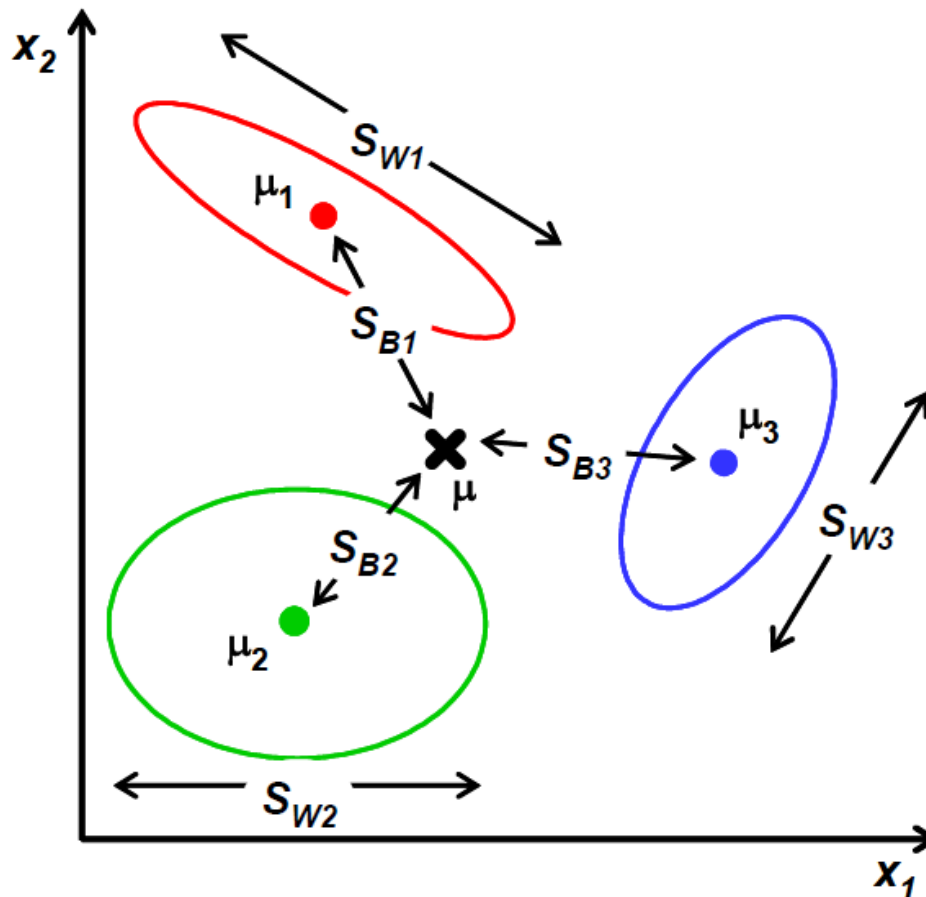
$$S_W^{-1} S_B w = \lambda w \quad \text{or directly } w = S_W^{-1} (\mu_1 - \mu_2)$$



线性判别分析(LDA)

- C类问题

$$J(w) = \frac{|\tilde{S}_B|}{|\tilde{S}_W|} = \frac{|w^T S_B w|}{|w^T S_W w|}$$



线性判别分析(LDA)

- **C类问题**

$$U = [X_1^1 \quad X_2^1 \quad X_1^2 \quad X_2^2 \quad \cdots \quad X_{n_c}^c]$$

$$S_w = \sum_{i=1}^c \sum_{j=1}^{n_c} (X_j^i - \mu_i)(X_j^i - \mu_i)^T$$

$$S_b = \sum_{i=1}^c (\mu_i - \mu)(\mu_i - \mu)^T$$

$$\frac{S_B}{S_W} V = \lambda V \qquad S_B V = \lambda S_B V$$

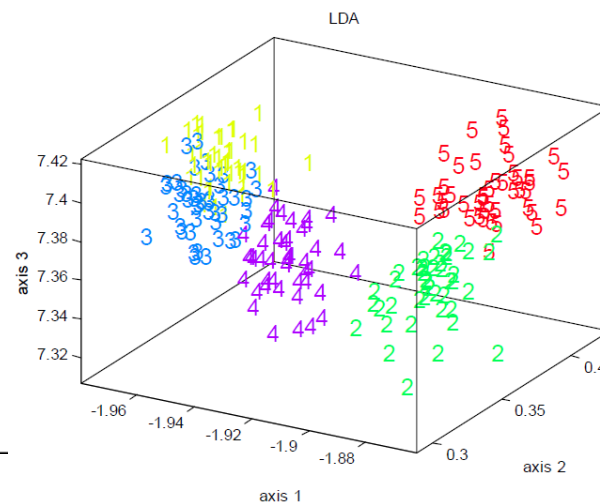
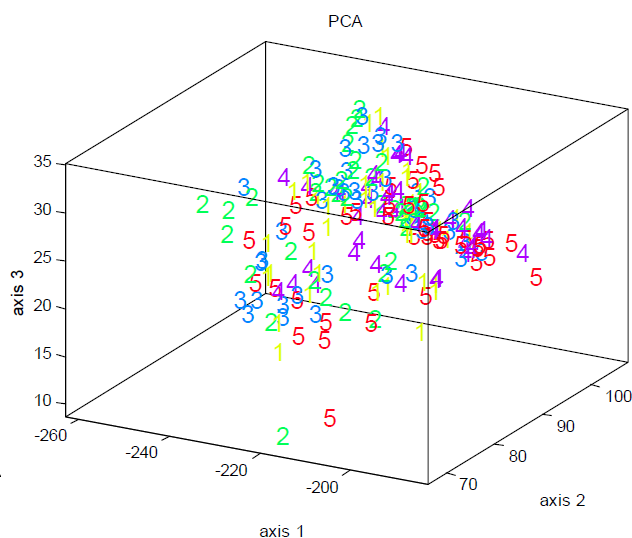
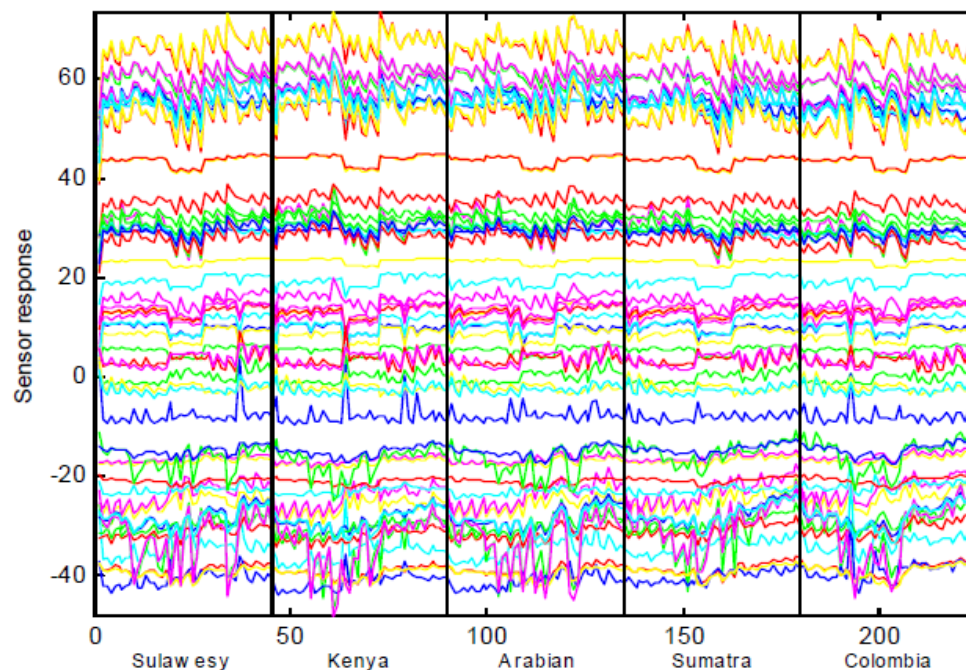
$$W = [V_1 \quad \cdots \quad V_k]$$

LDA的讨论

- 经FDA变换后，新的坐标系可能不是一个正交坐标系；
- 新的坐标维数最多为 $c-1$ ， c 为类别数；
- 只有当样本数足够多时，才能够保证类内散度矩阵 S_w 为非奇异矩阵（存在逆阵），而样本数少时 S_w 可能是奇异矩阵。
 - 存在向量 w ， $w^T S_w w = 0$
 - 如何解决？（PCA+LDA、Null Space LDA、VCA（ICML 2013 Best Paper））

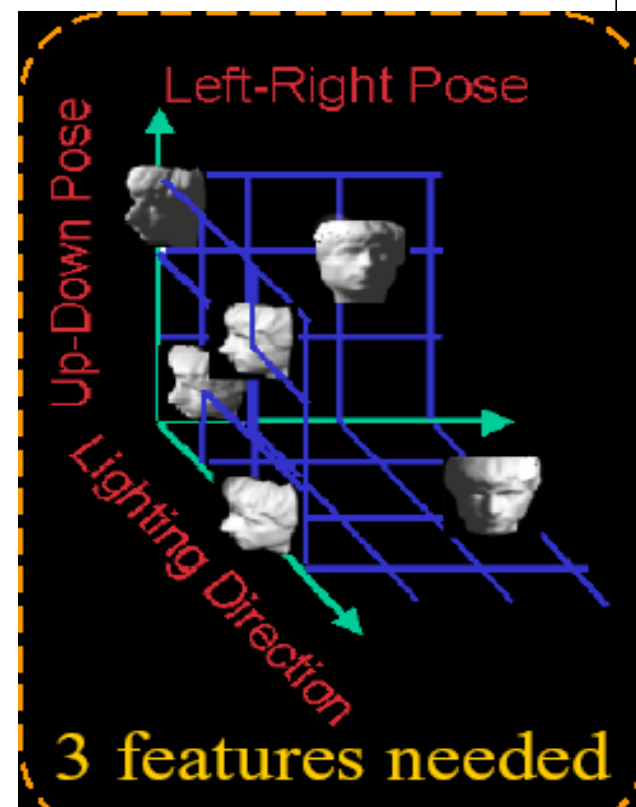
应用：咖啡豆分类

- 5种类型的咖啡豆
- 每类45个样本
- 60 维特征向量



2.3 流形学习：ISOMAP

- 问题：高维数据的低维结构
- 大脑中的数据存储方式



多维标度分析 (MDS)

校正矩阵: $P^e = I - \frac{1}{n}ee^T$

$D_{ij} = \left(\|x_i - x_j\|^2 \right)$: 距离矩阵

$$\Rightarrow \left(P^e D P^e \right)_{ij} = -2 \langle (x_i - \mu), (x_j - \mu) \rangle$$

问题: 给定 D , 如何选择相应的 x_i ?

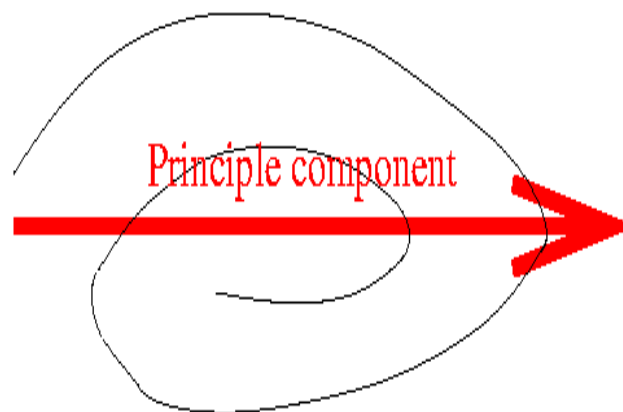
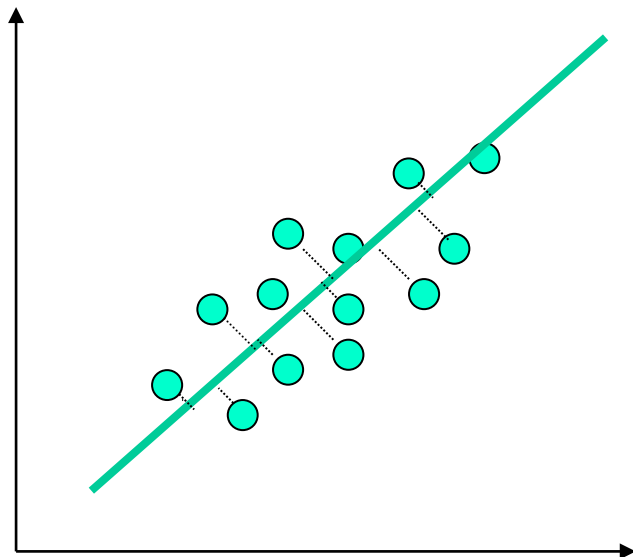
$$-\frac{P^e D P^e}{2} = \overline{D} = U_d \Sigma_d U_d^T = \left(U_d \Sigma_d^{0.5} \right) \left(U_d \Sigma_d^{0.5} \right)^T$$

$$U_d = [u_1, u_2, \dots, u_d], \quad \Sigma_d = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_d)$$

$$\Rightarrow x_i = \sqrt{\lambda_i} u_i, \text{ for } i = 1, \dots, d.$$

直接利用PCA

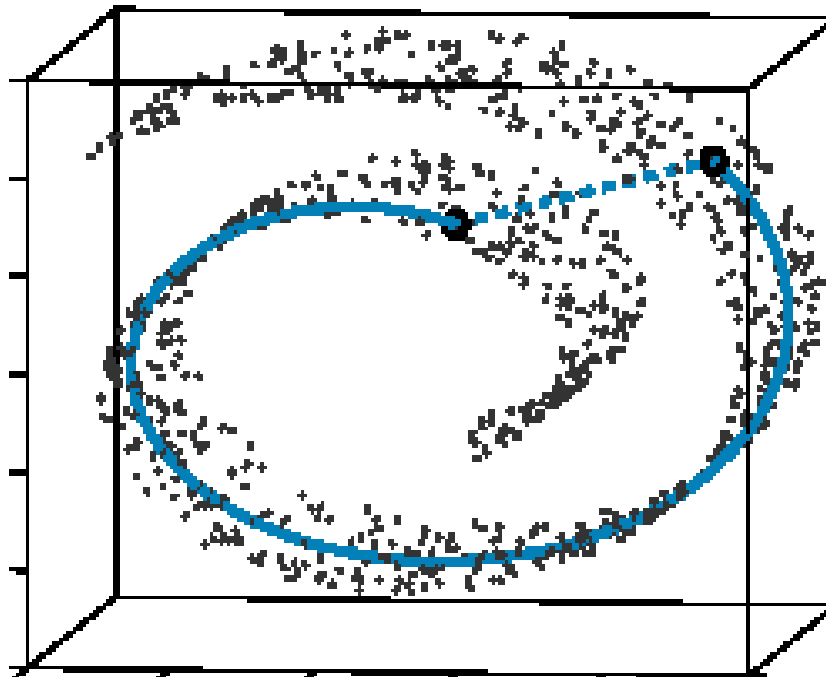
- 不能有效发现数据的低维结构



❖ Why?

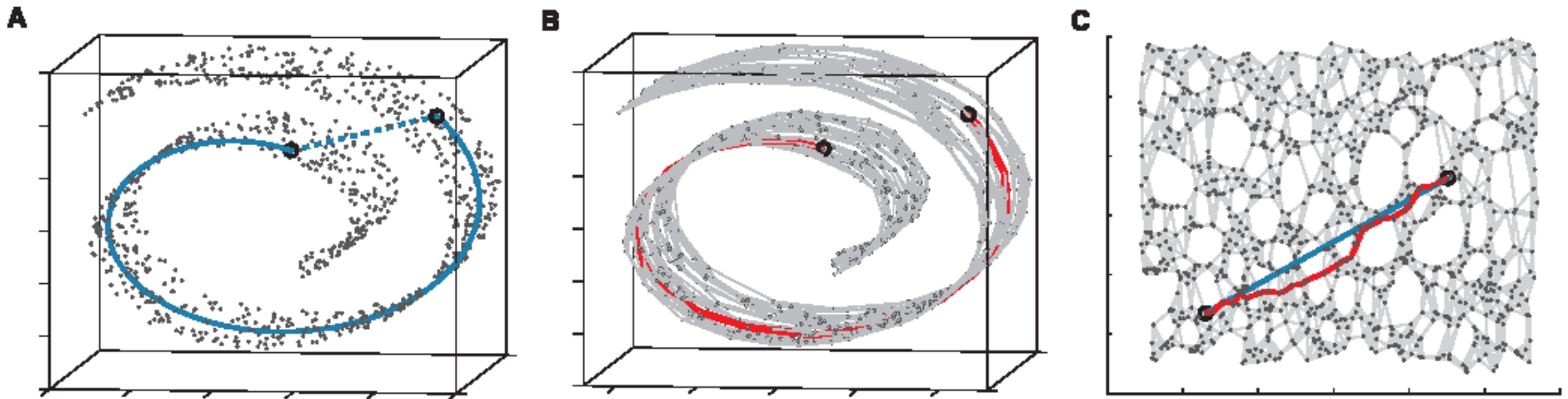
ISOMAP

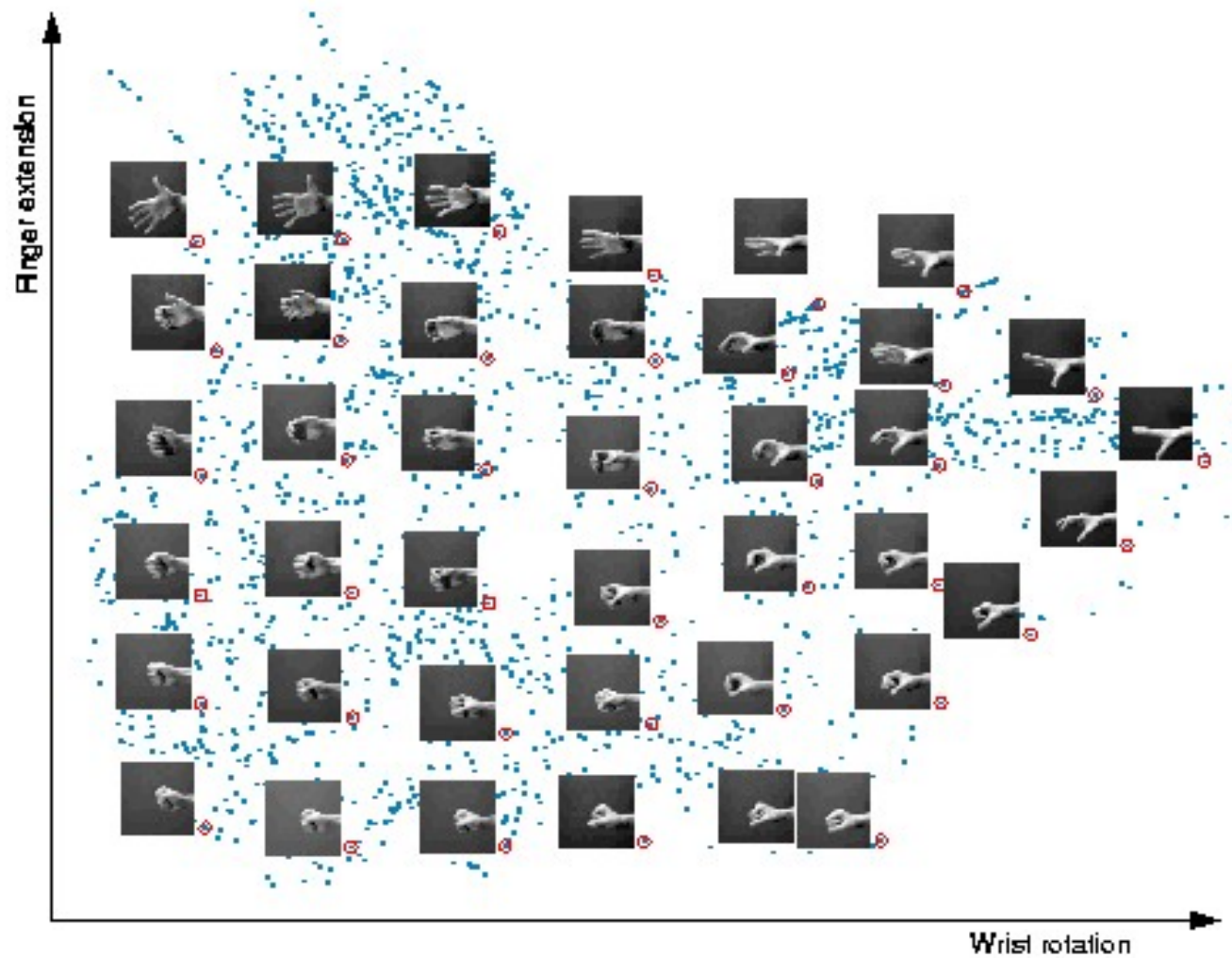
- PCA是一种基于欧氏距离的多维标度分析(Multi-dimensional scaling, MDS)
- 其它距离，如流形上的距离



ISOMAP

- 用流形上的距离代替欧氏距离，基于MDS算法，可得以下结果







其它流形学习方法

- 局部线性嵌入 (LLE)
- 拉普拉斯特征图 (Laplacian EigenMap)
- Hessian LLE
- ...

核特征提取方法：Kernel PCA

- 经典的PCA方法假设数据 \mathbf{x} 服从多变量高斯分布
- 但实际应用中这一假设可能不成立
- 基于核方法，我们可以将数据从原始空间 \mathbf{x} 转换为特征空间 $\phi(\mathbf{x})$ ，如果 $\phi(\mathbf{x})$ 服从高斯分布的话，我们可在特征空间下做PCA，即核PCA.
- 核PCA在高维甚至无穷维特征空间下进行主成分分析。庆幸的是，借助于核函数的性质， $\Phi(\mathbf{x})^t \Phi(\mathbf{y}) = K(\mathbf{x}, \mathbf{y})$ ，我们并不需要计算从原始空间到特征空间的显式映射。
 - 简要回顾非线性支持向量机

KPCA: 基本思想

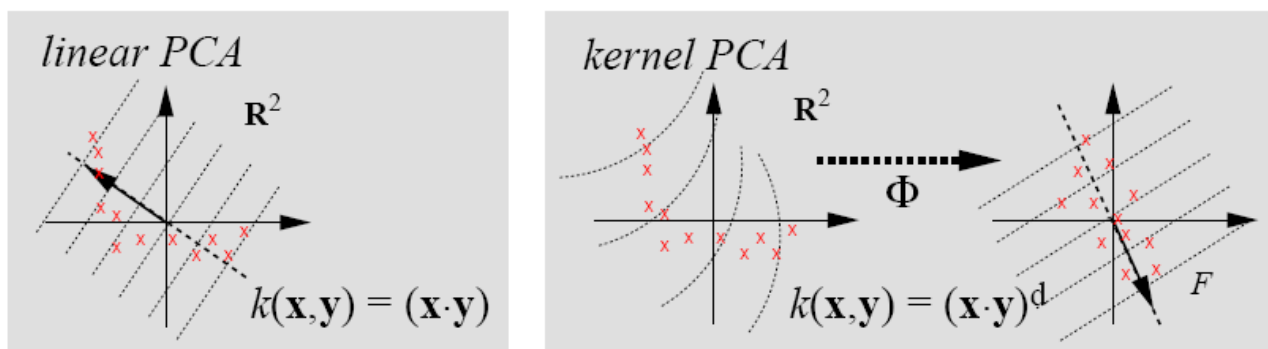


Fig. 1. Basic idea of kernel PCA: by using a nonlinear kernel function k instead of the standard dot product, we implicitly perform PCA in a possibly high-dimensional space F which is nonlinearly related to input space. The dotted lines are contour lines of constant feature value.

核PCA 问题表述

- 基础知识:
- 令 \mathbf{v} 为散度矩阵 $S = \sum_{i=1}^N \mathbf{x}_i \mathbf{x}_i^T$ 的一个特征向量
- 则 \mathbf{v} 应属于数据点 \mathbf{x}_i $i=1, 2, \dots, N$ 所组成的空间
- 证明:

$$S\mathbf{v} = \lambda\mathbf{v} \Rightarrow \mathbf{v} = \frac{1}{\lambda} \sum_{i=1}^N \mathbf{x}_i (\mathbf{x}_i^T \mathbf{v}) = \sum_{i=1}^N \alpha_i \mathbf{x}_i$$

核PCA 问题表述（续）

- 令 C 表示经过过去中心化映射 $\phi(\mathbf{x})$ 后的特征空间下的散度矩阵：

$$C = \sum_{i=1}^N \phi(\mathbf{x}_i) \phi(\mathbf{x}_i)^T$$

- 令 \mathbf{w} 为 C 的一个特征向量, 则 \mathbf{w} 也可以被写成线性组合形式:

$$\mathbf{w} = \sum_{k=1}^N \alpha_k \phi(\mathbf{x}_k)$$

- 同样, 有:

$$C\mathbf{w} = \lambda\mathbf{w}$$

- 综合上述, 可得:

$$\left(\sum_{i=1}^N \phi(\mathbf{x}_i) \phi(\mathbf{x}_i)^T \right) \left(\sum_{k=1}^N \alpha_k \phi(\mathbf{x}_k) \right) = \lambda \sum_{k=1}^N \alpha_k \phi(\mathbf{x}_k)$$

核PCA 问题表述（续）

$$\left(\sum_{i=1}^N \phi(\mathbf{x}_i) \phi(\mathbf{x}_i)^T\right) \left(\sum_{k=1}^N \alpha_k \phi(\mathbf{x}_k)\right) = \lambda \sum_{k=1}^N \alpha_k \phi(\mathbf{x}_k) \Rightarrow$$

$$\sum_{i=1}^N \sum_{k=1}^N \phi(\mathbf{x}_i) \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_k) \alpha_k = \lambda \sum_{k=1}^N \alpha_k \phi(\mathbf{x}_k) \Rightarrow$$

$$\sum_{i=1}^N \sum_{k=1}^N \phi(\mathbf{x}_l)^T \phi(\mathbf{x}_i) \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_k) \alpha_k = \lambda \sum_{k=1}^N \alpha_k \phi(\mathbf{x}_l)^T \phi(\mathbf{x}_k), l = 1, 2, \dots, N \Rightarrow$$

$$K^2 \boldsymbol{\alpha} = \lambda K \boldsymbol{\alpha} \Rightarrow$$

$$K \boldsymbol{\alpha} = \lambda \boldsymbol{\alpha}, \text{ 其中 } K_{ij} = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j).$$



核矩阵或 Gram 矩阵

Kernel PCA Formulation...

求解特征值问题 $K\mathbf{a} = \lambda\mathbf{a}$

然后将特征向量 \mathbf{w} 归一化为单位向量, 可得:

$$\|\mathbf{w}\|^2 = \left(\sum_{i=1}^N \alpha_i \phi(\mathbf{x}_i)\right)^T \left(\sum_{i=1}^N \alpha_i \phi(\mathbf{x}_i)\right) = \mathbf{a}^T K \mathbf{a} = 1 \Rightarrow$$

$$\mathbf{a}^T \mathbf{a} = \frac{1}{\lambda}$$

KPCA 算法

Step 1: 计算 Gram 矩阵: $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j), i, j = 1, \dots, N$

Step 2: 计算矩阵 \mathbf{K} 的 (特征值, 特征向量) 对: $(\boldsymbol{\alpha}^l, \lambda_l), l = 1, \dots, M$

Step 3: 特征向量归一化: $\boldsymbol{\alpha}^l \leftarrow \frac{\boldsymbol{\alpha}^l}{\lambda_l}$

因而, 矩阵 \mathbf{C} 的一个特征向量 \mathbf{w}^l 可被表示为: $\mathbf{w}^l = \sum_{k=1}^N \alpha_k^l \phi(\mathbf{x}_k)$

为投影测试特征向量 $\phi(\mathbf{x})$ 到 \mathbf{w}^l , 需要计算:

$$\phi(\mathbf{x})^T \mathbf{w}^l = \phi(\mathbf{x})^T \left(\sum_{k=1}^N \alpha_k^l \phi(\mathbf{x}_k) \right) = \sum_{k=1}^N \alpha_k^l k(\mathbf{x}_k, \mathbf{x})$$

不需要显式计算 ϕ !

特征映射中心化处理

到目前为止，我们假设特征向量 $\phi(\mathbf{x})$ 经过中心化处理，即 $\sum_{i=1}^N \phi(\mathbf{x}_i) = 0$

实际上，即使不满足上述假设，我们可以利用下述方式对Gram矩阵进行中心化处理

$$\tilde{K} = (I - 11^T / N) K (I - 11^T / N)$$

中心化处理后的核矩阵，即， $\sum_{i=1}^N \phi(\mathbf{x}_i) = 0$

我们可以使用类似的方式来处理测试样本，并进一步投影到特征向量空间
















































Scholkopf, Smola, Muller, “Nonlinear component analysis as a kernel eigenvalue problem,” Technical report #44, Max Plank Institute, 1996.

例：USPS 数字识别

❖ 邮政编码识别

❖ 自动投递

❖ 数字0-9

18  6→4	28  3→5	31  1→5	79  2→5	105  0→8	199  8→0	214  1→6	200  4→7
340  7→4	485  3→5	510  2→0	528  0→2	562  5→2	792  3→5	794  5→9	836  4→2
915  2→7	971  4→1	994  5→0	995  0→5	1007  0→2	1047  7→4	1105  3→2	1119  7→2
1307  3→5	1335  1→6	1354  7→4	1355  7→4	1358  3→5	1376  4→6	1380  4→9	1387  4→9
1417  8→2	1426  3→5	1432  3→5	1469  6→8	1529  7→2	1545  9→7	1657  5→8	1734  4→9
1814  1→4	1815  1→7	1816  1→4	1865  9→8	1872  4→7	1952  5→8	1978  5→3	

例：USPS 数字识别

# of components	Test Error Rate for degree (d)						
	1	2	3	4	5	6	7
32	9.6	8.8	8.1	8.5	9.1	9.3	10.8
64	8.8	7.3	6.8	6.7	6.7	7.2	7.5
128	8.6	5.8	5.9	6.1	5.8	6.0	6.8
256	8.7	5.5	5.3	5.2	5.2	5.4	5.4
512	n.a.	4.9	4.6	4.4	5.1	4.6	4.9
1024	n.a.	4.9	4.3	4.4	4.6	4.8	4.6
2048	n.a.	4.9	4.2	4.1	4.0	4.3	4.4

线性PCA

多项式核函数: $k(x, y) = (\mathbf{x}^T \mathbf{y})^d$

分类器: 线性 SVM

样本数目 $N = 3000$,

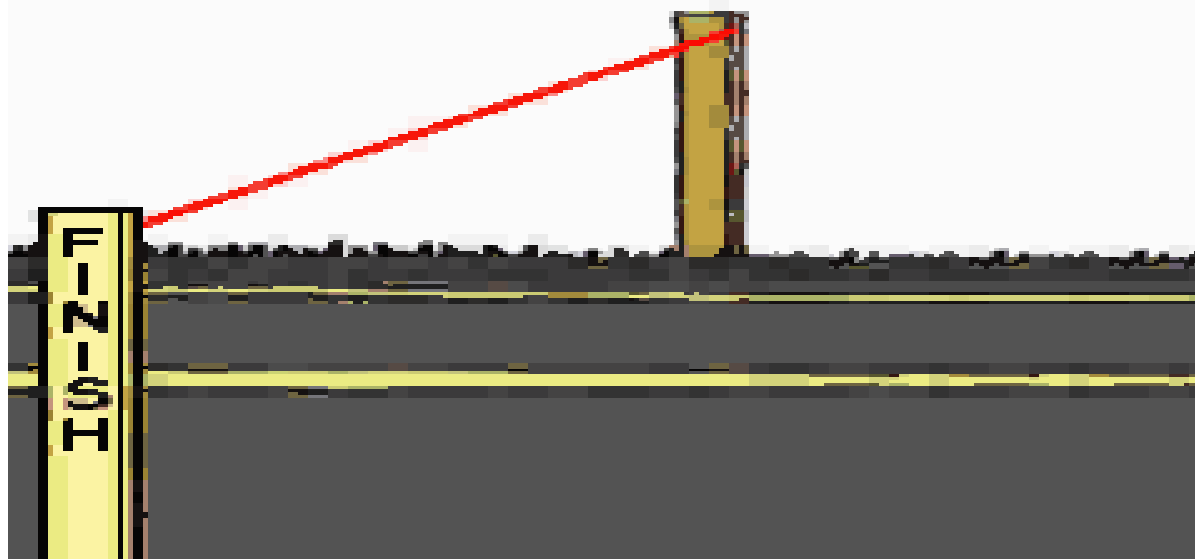
原始数据: 16 x16 图像

Scholkopf, Smola, Muller, "Nonlinear component analysis as a kernel eigenvalue problem," Technical report #44, Max Plank Institute, 1996.

其它核特征提取技术

- 核判别分析 (Kernel Fisher Discriminant Analysis)
- 核独立成分分析 (Kernel ICA)
- 核典型相关分析 (Kernel CCA)
-

Schölkopf: “*every (linear) algorithm that only use scalar (inner) products can implicitly be executed in Φ by using kernels, i.e. one can very elegantly construct a nonlinear version of a linear algorithm.*”



END