

机器学习与模式识别

第11讲 特征提取

2020~2021学年



特征提取

一、基本概念

二、类别可分性判据（特征评判标准）

三、主元分析（PCA，K-L变换）

四、基于类内散布矩阵的单类模式特征提取

五、特征提取实验示例

1 基本概念

■ 解决过拟合问题的主要手段

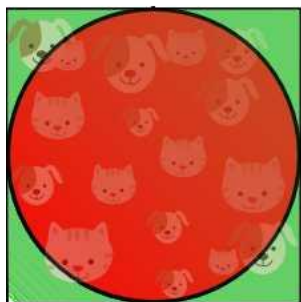
- 增加数据量

- 正则化

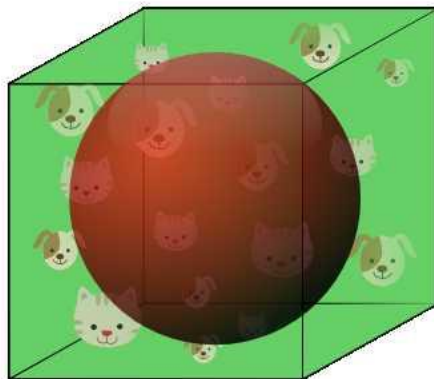
- 降维

■ 降维的思路源自维度灾难的问题

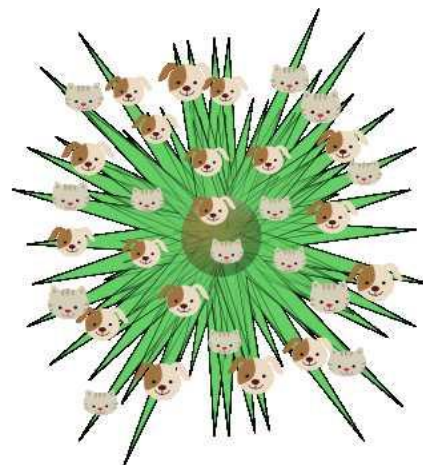
$$V_{\text{超球体}} = CR^n$$



$$V_{\text{超立方体}} = 2^n R^n$$



$$\lim_{n \rightarrow \infty} \frac{CR^n}{2^n R^n} = 0$$



1 基本概念

■ 特征提取

- 在模式识别领域，特征提取与选择尤为重要
- 基本思想是将处于高维空间中的原始样本特征描述映射为低维特征特征描述
- 不同的模式识别应用，需要采用不同的特征提取与选择方法

■ 以人脸识别为例



ORL (<https://cam-orl.co.uk/facedatabase.html>) 人脸数据库

1 基本概念

■ 以人脸识别为例

- 初始特征非常大，每幅图像的分辨率为 112×92
- 若将每个像素作为 1 维特征，则高达 10304
- 若把所有的原始特征都作为分类特征送到分类器，不仅分类器复杂，分类判别计算量大，且分类错误概率也不一定小

■ 解决思路

- 原始特征的特征空间有很大的冗余，可以用很小的空间相当好地近似表示图像，这一点与压缩的思想类似
- 因此有必要减少特征数目，以获取“少而精”的分类特征，即获取特征数目少且能使分类错误概率小的特征向量

对特征的要求

■ 作为识别分类用的特征应具备以下几个条件:

① 具有很大的识别信息量

特征应具有很好的可分性，使分类器容易判别。

② 具有可靠性

去掉模棱两可，似是而非，对判别作用不大的特征。

③ 具有尽可能强的独立性

特征之间的强相关性并没有增加分类信息，反而给问题的数值求解带来病态条件。

④ 数量尽可能少，同时损失的信息尽量小

模式识别中减少特征数目的方法有两种：一种是**特征提取**，另一种是**特征选择**。

- **原始特征**：通过直接测量得到的特征称为原始特征。比如人体的各种生理指标（描述其健康状况）；数字图像中的各像素点的亮度值（描述图像内容），都是原始特征。
- **特征提取**：通过映射（变换）的方法把高维的特征向量转为低维的特征向量。

$$A: X \rightarrow Y$$

- **特征选择**：从原始特征中挑选出一些最有代表性、分类性能好的特征，以达到降低特征空间维数的目的。

$$D \text{ 个特征} \rightarrow d \text{ 个特征} (d < D)$$

特征提取与具体问题有很大关系，没有理论能给出对任何问题都有效的特征提取方法。

在许多实际问题中，那些最重要的特征往往不易找到，使得特征选择和特征提取成为构造模式识别系统最困难的任务之一。

1、用于分类的模式特征的特点

在模式识别过程中，模式特征的确定比较复杂。

为有效进行模式识别，选择的特征应满足以下条件：

(1) 特征既可以是直接测量的，也可是常规变换后的。

观察对象的原始特征应该能方便地被设备或人工采集，并能以适合后续处理的方式输入到计算机中。

特征既可以是直接测量得到数据，也可以是在测量数据基础上进行常规变换或分析后形成的二次特征（如图像分割后的子目标特性表达，语音信号的时频谱等）。

(2) 类内有凝聚性。 选择的特征对同一类应具有稳定性。

由于模式类是由具有相似特性的若干个样本构成的，因此它们同属一类，首要前提是特性相似，反映在取值分布上，应该有较稳定的凝聚性。

(3) 类间有差异性。 选择的特征对不同的类应该有可辨识的差异。若不同类的模式的特征值差异很小，则说明所选择的特征对于不同的类没有什么差异，作为分类的依据时，容易使不同的类产生混淆，使误识率增大。一般来讲。**特征的 类间差异 应该大于 类内差异。**

2、特征的类别

(1) 物理特征（观测型特征）

物理特征是比较直接、人们容易感知的特征，一般在设计模式识别系统时容易被选用。如为了描述指定班级中的某个学生，可以用以下物理特征：性别、身高、胖瘦、肤色等外在特征。物理特征虽然容易感知，却未必能非常有效地表征分类对象。

(2) 结构特征（几何型特征）

结构特征是指反映对象表观轮廓或逻辑结构的一类特征，既包括对象组分之间的具象的几何邻接关系，也包括对象组分之间抽象的相互作用关系。一般是先将对象分割成若干个基本组分，再确定**组分间的各种邻接或作用关系**。**结构特征的表达能力一般要高于物理特征**，如指纹识别、汉字识别的成功都离不开结构特征的选择。

结构特征对对象的尺寸往往不太敏感。

结构特征比物理特征要抽象一些，能较好反映对象的物理轮廓与光学属性特征（如人的指纹特征、人脸的五官结构信息等），也能反映对象内部抽象但可定义的交互作用关系（如社会网络中的社团发现、重要节点辨识等）。

(3) 符号特征（编号型特征）

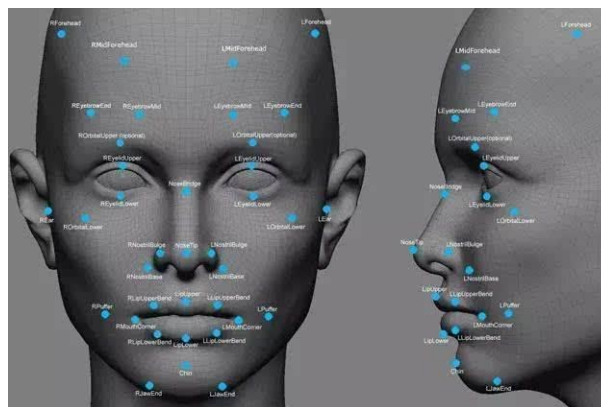
一般来说，符号特征是为了表征观察对象而设立的特征，如给每个学生设立一个学号，作为标志每个学生的特征。由于学号是人为设定的，可以保证唯一性，但这种特征是抽象的，不容易被人感知。符号特征有时和观察对象的固有特性没有任何联系，有时则是对物理或结构特征的某种转换的结果（如：基于注册时间赋予编号）。

3、特征的形成

在设计一个具体的模式识别系统时，往往是先接触一些训练样本，由领域专家和系统工程师联合研究模式类所包含的特征信息，并给出相应的表述方法。这一阶段的主要目标是获取尽可能多的表述特征。在这些特征中，有些可能满足**类内凝聚、类间差异**的要求，有的则可能不满足，不能作为分类的依据。根据样例分析得到一组表述观察对象的特征值，而不论特征是否实用，称这一步为**特征形成**，得到的特征称为**原始特征**。

在保证性能的条件下，通过降低特征空间的维数来减少分类方法的复杂度。

- 特征提取
- 特征选择



特征提取和特征选择都不考虑针对具体应用需求的原始特征形成过程，而是假设原始特征形成工作已经完成。

特征形成是模式识别过程中的重点和难点之一

4、特征提取和特征选择

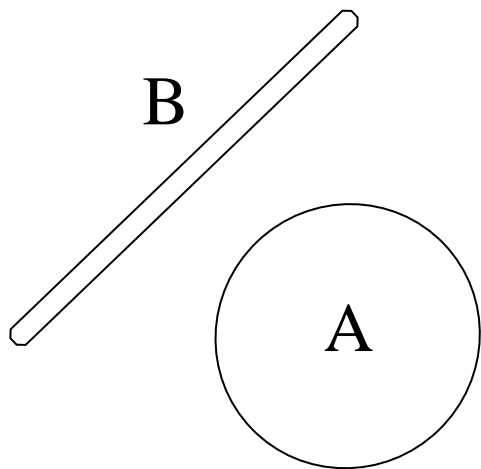
特征选择是指从原始特征中挑选出对一组分类最有利于分类的特征，

特征提取是指通过变换的方法获取最有利于分类的一组特征。变换后的特征称为**二次特征**，它们是原始特征的某种组合，最常用的是线性组合。

如果选择或提取的特征数相对原特征数有大幅减少，则特征选择也同时实现了**特征降维**。

要求：当特征向量在特征空间中发生移动、旋转、缩放时，模式类内部的凝聚性以及模式类间的差异性应该能保持。

【 Toy example 】 特征选择与特征提取的区别： 对一个条形和圆进行识别。



解：[方法1]

① 特征抽取： 测量三个结构特征

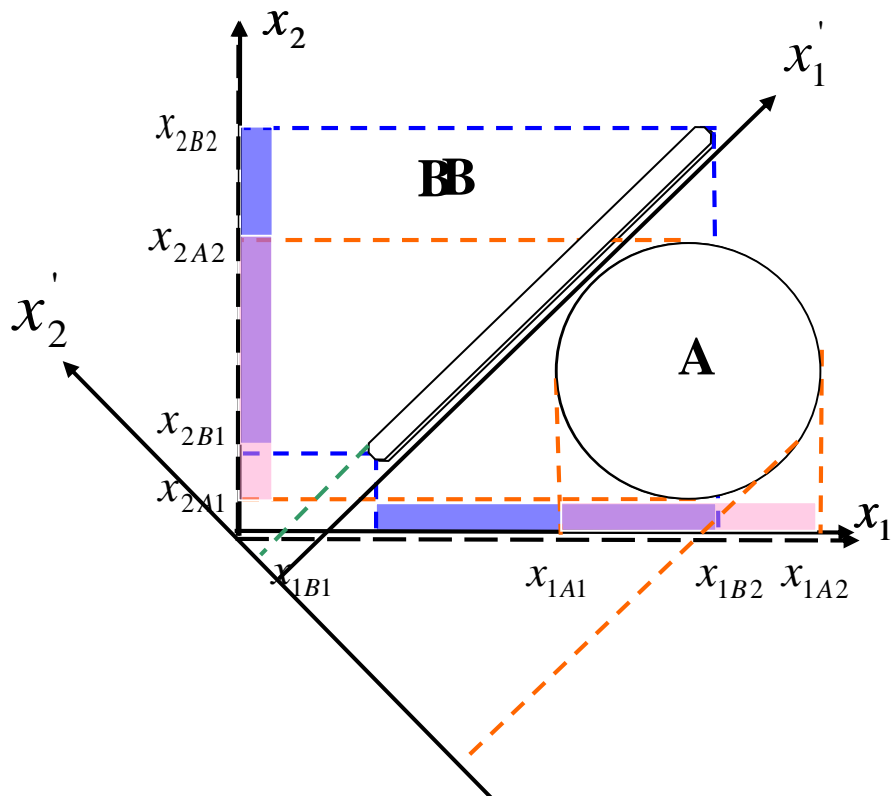
(a) 周长

(b) 面积

(c) 两个互相垂直的内径比

② 分析： (c)是具有分类能力的特征， 故选(c)，
扔掉(a)、 (b)。

—— 特征选择： 一般针对待分类对象的物理特征或结构特征进行选择。



[方法2]: ① 特征抽取: 测量物体向两个坐标轴的投影值, 则A、B各有2个值域区间。可以看出, 两个物体的投影有重叠, 直接使用投影值无法将两者区分开。

② 分析: 将坐标系按逆时针方向做一旋转变换, 或物体按顺时针方向变, 并适当平移等。根据物体在 x_2' 轴上投影的坐标值的正负可区分两个物体。

——特征提取, 一般是基于某个优化指标 (可分性), 采用数学变换方法进行。

特征提取和特征选择，都是在不降低或很少降低分类性能的前提下，降低特征空间维数，主要好处：

(1) **简化计算**。特征空间的维数越高，需占用的计算机资源越多，设计和计算也就越复杂。

(2) **简化特征空间结构**。由于**特征提取和选择是去除类间差别小的特征，保留类间差别大的特征**，因此，在特征空间中，每类所占据的子空间结构可分离性更强，从而也简化了类间分界面形状的复杂度。

(3) **降低训练样本量**。

二、类别可分性判据（特征评判标准）

- ◆ 衡量不同特征及其组合对分类是否有效的定量准则
- ◆ 理想的准则：使分类器错误概率最小
- ◆ 类别可分性判据应满足的条件：
 - 与错误率有单调关系（可分性判据越大，误分率越小）
 - 距离特性： $J_{ij} > 0$, if $i \neq j$; $J_{ij} = 0$, if $i = j$; $J_{ij} = J_{ji}$
 - 独立特征的可加性： $J_{ij}(x_1, x_2, \dots, x_d) = \sum_{k=1}^d J_{ij}(x_k)$
 - 特征数目的单调性： $J_{ij}(x_1, x_2, \dots, x_d) \leq J_{ij}(x_1, x_2, \dots, x_d, x_{d+1})$
- ◆ 常见类别可分性判据：
 - 基于距离度量
 - 基于概率分布
 - 基于熵函数

2.1 基于距离的可分性判据

基于距离的可分性判据直接依靠样本计算，直观简洁，物理概念清晰，因此目前应用较为广泛。基于距离的可分性判据的出发点是：**各类样本之间的距离越大、类内离散度越小，则类别的可分性越好**。常用的点间距离有：欧氏距离、马氏距离、绝对距离（城市距离、Hamming距离）、Minkowsky距离等。以下取欧式距离：

1. 类内散布距离和类内散布矩阵

1) ω_i 类的**类内距离一**：**类内（样本间）均方两两距离**。

$$\overline{D_i^2} = E[\| \mathbf{X}_k - \mathbf{X}_l \|^2 | \omega_i] = E[(\mathbf{X}_k - \mathbf{X}_l)^T (\mathbf{X}_k - \mathbf{X}_l) | \omega_i]$$

其中 \mathbf{X}_k 和 \mathbf{X}_l 均为同一类模式样本。

若 $\{\mathbf{X}_i\}$ 中的样本相互独立，有

$$\begin{aligned}\overline{D_i^2} &= E[(\mathbf{X}_k^T \mathbf{X}_k - \mathbf{X}_k^T \mathbf{X}_l - \mathbf{X}_l^T \mathbf{X}_k + \mathbf{X}_l^T \mathbf{X}_l) | \omega_i] \\ &= 2E[\mathbf{X}_k^T \mathbf{X}_k] - 2E[\mathbf{X}_k^T]E[\mathbf{X}_l] = 2\text{Tr}[\mathbf{R}_i - \mathbf{M}_i \mathbf{M}_i^T] \\ \overline{D_i^2} &= E[(\mathbf{X}_k - \mathbf{M}_i + \mathbf{M}_i - \mathbf{X}_l)^T (\mathbf{X}_k - \mathbf{M}_i + \mathbf{M}_i - \mathbf{X}_l) | \omega_i] \\ &= E[(\mathbf{X}_k - \mathbf{M}_i)^T (\mathbf{X}_k - \mathbf{M}_i) + (\mathbf{X}_l - \mathbf{M}_i)^T (\mathbf{X}_l - \mathbf{M}_i) | \omega_i] \\ &= 2\text{Tr}[\mathbf{C}_i] = 2 \sum_{k=1}^n \sigma_k^2\end{aligned}$$

式中， \mathbf{R}_i ：第*i*类的自相关矩阵；

\mathbf{M}_i ：第*i*类的均值向量； \mathbf{C}_i ：第*i*类的协方差矩阵；

σ_k^2 ： \mathbf{C}_i 主对角线上的元素，表示模式向量第*k*个分量的方差；

Tr：矩阵的迹（方阵主对角线上各元素之和）。

2) ω_i 类的类内距离二：类内（样本间）均方散布距离。

$$\overline{D_{w,i}^2} = E[\|\mathbf{X}_k - \mathbf{M}_i\|^2 | \mathbf{X}_k \in \omega_i]$$

可证： $\overline{D_i^2} = 2\overline{D_{w,i}^2}$ ——有2倍关系，因为前者有重复计算。

3) 类内距离的计算式（样本版本）。

$$\overline{D_{w,i}^2} = E[\| \mathbf{X}_k - \mathbf{M}_i \|^2 | \omega_i] = \frac{1}{n_i} \sum_{k=1}^{n_i} (\mathbf{X}_k^i - \mathbf{M}_i)^T (\mathbf{X}_k^i - \mathbf{M}_i)$$

$$\overline{D_w^2} = \sum_{i=1}^c P(\omega_i) \overline{D_{w,i}^2} = \sum_{i=1}^c P(\omega_i) \left[\frac{1}{n_i} \sum_{k=1}^{n_i} (\mathbf{X}_k^i - \mathbf{M}_i)^T (\mathbf{X}_k^i - \mathbf{M}_i) \right]$$

$$\overline{D_i^2} = E[(\mathbf{X}_k - \mathbf{X}_l)^T (\mathbf{X}_k - \mathbf{X}_l) | \omega_i] = \frac{1}{n_i(n_i - 1)} \sum_{k=1}^{n_i} \sum_{\substack{l=1 \\ l \neq k}}^{n_i} (\mathbf{X}_k^i - \mathbf{X}_l^i)^T (\mathbf{X}_k^i - \mathbf{X}_l^i)$$

4) 类内散布矩阵：表示各样本点**围绕类均值的散布**情况，
以及各特征间互相关情况——即第*i*类分布的协方差矩阵。

$$\mathbf{S}_{w,i} = E[(\mathbf{X} - \mathbf{M}_i)(\mathbf{X} - \mathbf{M}_i)^T | \omega_i]$$

$$= \frac{1}{n_i} \sum_{k=1}^{n_i} (\mathbf{X}_k^i - \mathbf{M}_i)(\mathbf{X}_k^i - \mathbf{M}_i)^T \quad \text{显然:} \quad \overline{D_{w,i}^2} = \text{Tr}(\mathbf{S}_{w,i})$$

特征选择和提取，应使类内散布矩阵的迹 愈小愈好。

2. 类间散布距离和类间散布矩阵

- 1) **类间距离一**：各模式类均值向量之间**距离平方的加权和——类间均方两两距离**，记为 \overline{D}^2 。

$$\overline{D}^2 = \frac{1}{2} \sum_{i=1}^c P(\omega_i) \sum_{j=1}^c P(\omega_j) \| \mathbf{M}_i - \mathbf{M}_j \|^2 \quad \text{可证:}$$

$$\overline{D}^2 = \sum_{i=1}^c P(\omega_i) \| \mathbf{M}_i - \mathbf{M}_0 \|^2 = \sum_{i=1}^c P(\omega_i) (\mathbf{M}_i - \mathbf{M}_0)^T (\mathbf{M}_i - \mathbf{M}_0)$$

- 2) **类间距离二**：各模式类均值向量与总体均值向量之间**距离平方的加权和——类间均方散布距离**，记为 D_b^2 （见上式）。

式中， $P(\omega_i)$ ： ω_i 类的先验概率； \mathbf{M}_i ： ω_i 类的均值向量；

\mathbf{M}_0 ：所有 c 类模式的总体均值向量。

$$\mathbf{M}_0 = E[\mathbf{X}] = E_{\omega} E[\mathbf{X} / \omega = \omega_i] = \sum_{i=1}^c P(\omega_i) \mathbf{M}_i$$

- 3) 类间散布矩阵：表示 c 类模式在空间的散布情况，记为 \mathbf{S}_b 。

$$\mathbf{S}_b = \sum_{i=1}^c P(\omega_i) (\mathbf{M}_i - \mathbf{M}_0) (\mathbf{M}_i - \mathbf{M}_0)^T$$

注意：与类间距离的转置位置不同。

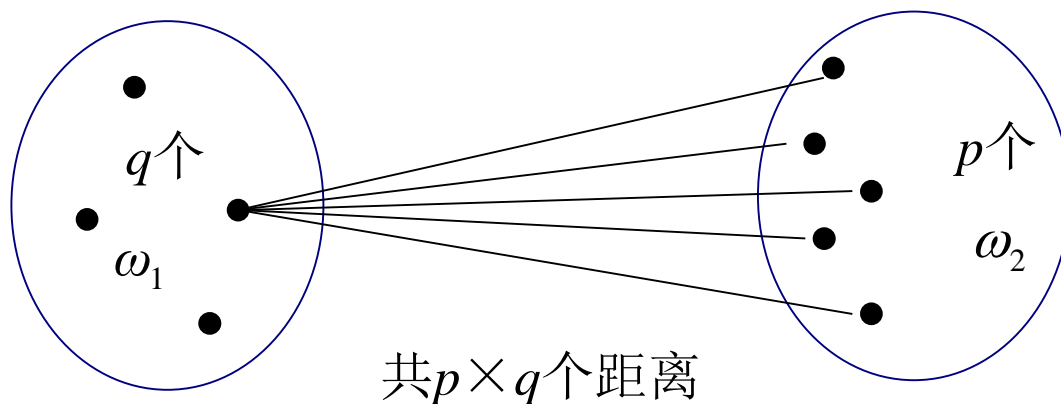
- 4) 类间散布距离与类间散布矩阵的关系： $\overline{D}_b^2 = \text{Tr}(\mathbf{S}_b)$

类间散布矩阵的迹愈大愈有利于分类。

3. 多类模式向量间的总体距离和总体散布矩阵

1) 两类情况的距离

设 ω_1 类中有 q 个样本， ω_2 类中有 p 个样本。



两个类之间的距离 = $p \times q$ 个点间距离的平均值

类似地 \downarrow 多类情况

多类间任意两点间两两距离的平均值

\downarrow
多类间任意两点间两两距离平方的平均值

2) 多类情况的总体距离

任意类的组合

特定两类间
任意样本的组合

(1) 多类模式向量间的**总体平均距离** J_d

$$J_d = \frac{1}{2} \sum_{i=1}^c P(\omega_i) \sum_{j=1}^c P(\omega_j) \frac{1}{n_i n_j} \sum_{k=1}^{n_i} \sum_{l=1}^{n_j} D^2(\mathbf{X}_k^i, \mathbf{X}_l^j) \quad (5-8)$$

式中, $P(\omega_i)$ 和 $P(\omega_j)$: ω_i 和 ω_j 类先验概率; c : 类别数;

\mathbf{X}_k^i : ω_i 类的第 k 个样本; \mathbf{X}_l^j : ω_j 类的第 l 个样本;

n_i 和 n_j : ω_i 和 ω_j 类的样本数;

$D^2(\mathbf{X}_k^i, \mathbf{X}_l^j)$: \mathbf{X}_k^i 和 \mathbf{X}_l^j 间欧氏距离的平方。

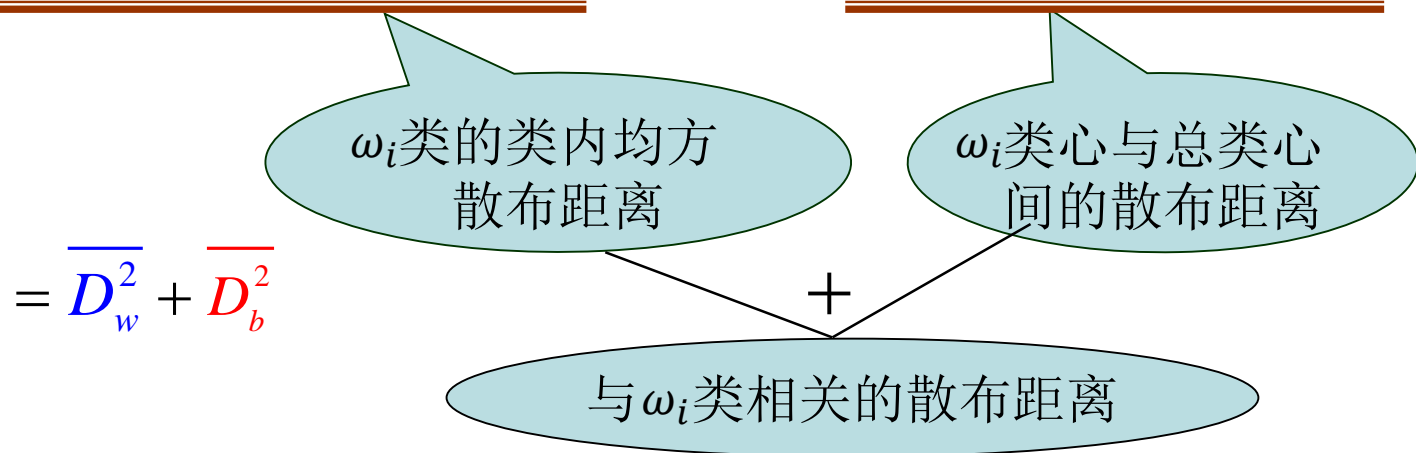
(2) J_d 的另一种形式: 将以下三式代入(5-8)式

距离取欧式距离: $D^2(\mathbf{X}_k^i, \mathbf{X}_l^j) = (\mathbf{X}_k^i - \mathbf{X}_l^j)^T (\mathbf{X}_k^i - \mathbf{X}_l^j) \quad (5-9)$

ω_i 类的均值向量: $\mathbf{M}_i = \frac{1}{n_i} \sum_{k=1}^{n_i} \mathbf{X}_k^i \quad (5-10)$

所有类的总体均值向量: $\mathbf{M}_0 = \sum_{i=1}^c P(\omega_i) \mathbf{M}_i \quad (5-11)$

$$J_d = \sum_{i=1}^c P(\omega_i) \left[\frac{1}{n_i} \sum_{k=1}^{n_i} (\mathbf{X}_k^i - \mathbf{M}_i)^T (\mathbf{X}_k^i - \mathbf{M}_i) \right] + \sum_{i=1}^c P(\omega_i) \left[(\mathbf{M}_i - \mathbf{M}_0)^T (\mathbf{M}_i - \mathbf{M}_0) \right]$$



多类模式向量之间的总均方距离=各类均方距离的先验概率加权和

多类模式向量之间的总体均方距离= { 模式类内散布距离加权和
+
模式类间散布距离加权和

3) 多类情况的散布矩阵

多类类间散布矩阵:

$$S_b = \sum_{i=1}^c P(\omega_i) (\mathbf{M}_i - \mathbf{M}_0) (\mathbf{M}_i - \mathbf{M}_0)^T$$

ω_i 类内散布矩阵: $S_{w,i} = E[(X - M_i)(X - M_i)^T | \omega_i]$

$$= \frac{1}{n_i} \sum_{k=1}^{n_i} (X_k^i - M_i)(X_k^i - M_i)^T \quad \text{【计算式】}$$

多类类内散布矩阵:

$$S_w = \sum_{i=1}^c P(\omega_i) S_{w,i} = \sum_{i=1}^c P(\omega_i) E[(X - M_i)(X - M_i)^T | \omega_i]$$

$$= \sum_{i=1}^c P(\omega_i) \frac{1}{n_i} \sum_{k=1}^{n_i} (X_k^i - M_i)(X_k^i - M_i)^T \quad \text{【计算式】}$$

—— 各类模式协方差矩阵的先验加权平均值。

多类总体散布矩阵:

$$S_t = E[(X - M_0)(X - M_0)^T] = S_b + S_w$$

4) 多类模式的总体均方距离 J_d 与总体散布矩阵 S_t 的关系

$$J_d = \text{tr}(S_t) = \text{tr}(S_b + S_w)$$

2.2 基于概率分布的可分性测度

1. 散度

1) 散度的定义

出发点：对数似然比 含有 类别的可分性信息。

设 ω_i, ω_j 类的概率密度函数分别为 $p(\mathbf{X} | \omega_i)$ 和 $p(\mathbf{X} | \omega_j)$

$$\omega_i \text{ 类对 } \omega_j \text{ 类的对数似然比: } l_{ij} = \ln \frac{p(\mathbf{X} | \omega_i)}{p(\mathbf{X} | \omega_j)}$$

$$\omega_j \text{ 类对 } \omega_i \text{ 类的对数似然比: } l_{ji} = \ln \frac{p(\mathbf{X} | \omega_j)}{p(\mathbf{X} | \omega_i)}$$

对不同的 X ，似然函数不同，对数似然比体现的可分性不同，通常采用平均可分性信息——对数似然比的期望值。

ω_i 类对数似然比的期望值：

$$E\{x\} = \int_{-\infty}^{\infty} xp(x)d(x)$$

$$I_{ij} = E[l_{ij}] = \int_X p(\mathbf{X}|\omega_i) \ln \frac{p(\mathbf{X}|\omega_i)}{p(\mathbf{X}|\omega_j)} d\mathbf{X}$$

ω_j 类对数似然比的期望值：

$$I_{ji} = E[l_{ji}] = \int_X p(\mathbf{X}|\omega_j) \ln \frac{p(\mathbf{X}|\omega_j)}{p(\mathbf{X}|\omega_i)} d\mathbf{X}$$

散度等于两类间对数似然比期望值之和。

ω_i 类对 ω_j 类的散度定义为 J_{ij} ：

$$J_{ij} = I_{ij} + I_{ji} = \int_X [p(\mathbf{X}|\omega_i) - p(\mathbf{X}|\omega_j)] \ln \frac{p(\mathbf{X}|\omega_i)}{p(\mathbf{X}|\omega_j)} d\mathbf{X}$$

散度表示了区分 ω_i 类和 ω_j 类的总的平均信息。

——特征选择和特征提取应使散度尽可能的大。

2) 散度的性质

(1) $J_{ij} = J_{ji}$

$$J_{ij} = I_{ij} + I_{ji} = \int_X [p(\mathbf{X}|\omega_i) - p(\mathbf{X}|\omega_j)] \ln \frac{p(\mathbf{X}|\omega_i)}{p(\mathbf{X}|\omega_j)} d\mathbf{X}$$

$$J_{ji} = I_{ji} + I_{ij} = \int_X [p(\mathbf{X}|\omega_j) - p(\mathbf{X}|\omega_i)] \ln \frac{p(\mathbf{X}|\omega_j)}{p(\mathbf{X}|\omega_i)} d\mathbf{X}$$

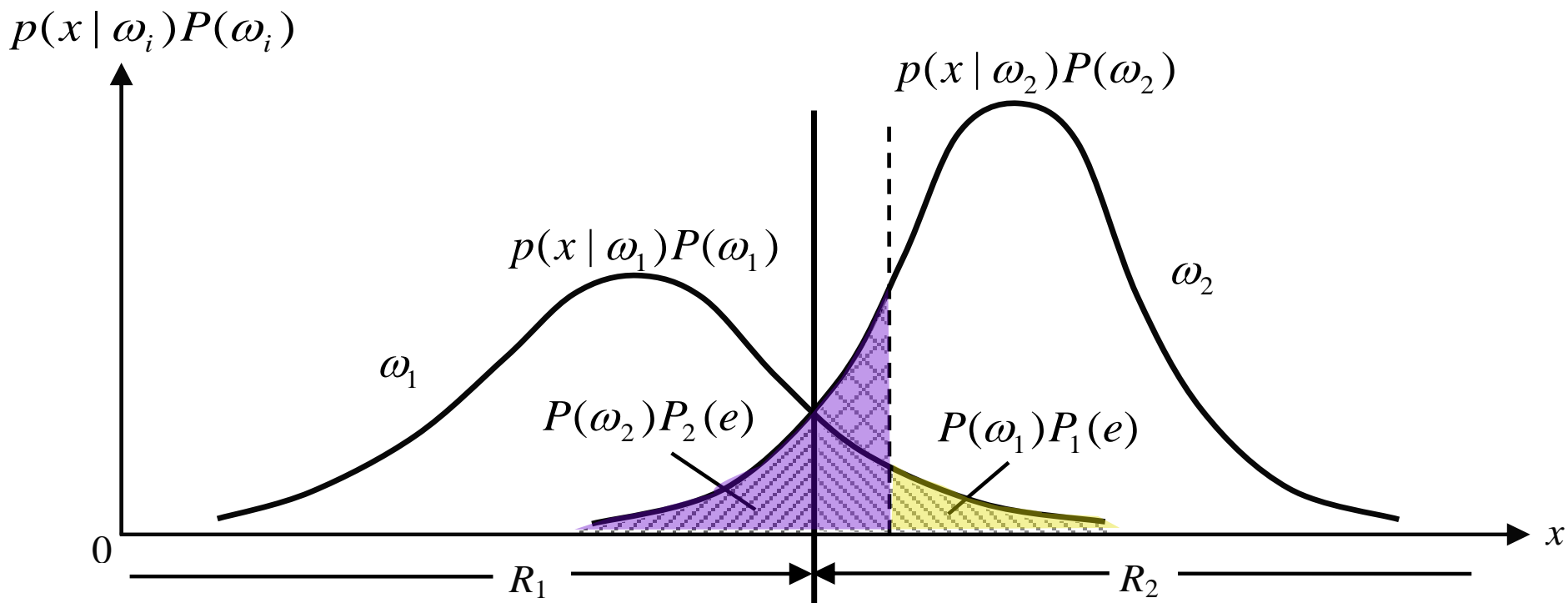
(2) J_{ij} 为非负, 即 $J_{ij} \geq 0$ 。

当 $p(\mathbf{X} | \omega_i) \neq p(\mathbf{X} | \omega_j)$ 时, $J_{ij} > 0$,

$p(\mathbf{X} | \omega_i)$ 与 $p(\mathbf{X} | \omega_j)$ 相差愈大, J_{ij} 越大。

当 $p(\mathbf{X} | \omega_i) = p(\mathbf{X} | \omega_j)$, 两类分布密度相同, $J_{ij} = 0$ 。

(3) 错误率分析中，两类概率密度曲线交叠越少，错误率越小。



由散度的定义式 $J_{ij} = I_{ij} + I_{ji} = \int_X [p(\mathbf{X}|\omega_i) - p(\mathbf{X}|\omega_j)] \ln \frac{p(\mathbf{X}|\omega_i)}{p(\mathbf{X}|\omega_j)} d\mathbf{X}$

可知，散度愈大，两类概率密度函数曲线相差愈大，交叠愈少，分类错误率愈小。

(4) 散度具有独立可加性：对于模式向量 $\mathbf{X} = [x_1, x_2, \dots, x_n]^T$ ，若各分量相互独立，则有【大家可以试证之】

$$J_{ij}(\mathbf{X}) = J_{ij}(x_1, x_2, \dots, x_n) = \sum_{k=1}^n J_{ij}(x_k)$$

据此可估计每一个特征在分类中的重要性：

散度较大的特征含有较大的可分信息——保留。

(5) 独立可加性表明：增加独立新特征，散度增加（意味着什么？）

$$J_{ij}(x_1, x_2, \dots, x_n) \leq J_{ij}(x_1, x_2, \dots, x_n, x_{n+1})$$

3) 两个等方差正态分布模式类的散度

设 ω_i 类和 ω_j 类的概率密度函数分别为

$$p(\mathbf{X}|\omega_i) \sim N(\mathbf{M}_i, \mathbf{C})$$

$$p(\mathbf{X}|\omega_j) \sim N(\mathbf{M}_j, \mathbf{C})$$

可得到 ω_i 类对 ω_j 类的散度为

$$J_{ij} = \text{Tr}[(\mathbf{C}^{-1}(\mathbf{M}_i - \mathbf{M}_j)(\mathbf{M}_i - \mathbf{M}_j)^{\text{T}})] = (\mathbf{M}_i - \mathbf{M}_j)^{\text{T}} \mathbf{C}^{-1}(\mathbf{M}_i - \mathbf{M}_j)$$

——正好等于两模式类均值向量间的**马氏距离的平方**

以一维正态分布为例：

$$J_{ij} = \frac{(m_i - m_j)^2}{\sigma^2}$$

两类均值向量间
距离越远，散度愈大

每类的类内向量
分布愈集中（方差越
小），散度愈大

三、主元分析（PCA，K-L变换）

无监督降维

主成分分析（PCA，Principal Components Analysis）

是一种有效的特征线性变换，也称为**K-L变换/Hotelling变换**，
是一种基于目标统计特性的最佳正交变换，**最佳性体现在变换后产生的新的分量正交或不相关。**

一个中心：原始特征空间的重构（将相关变成不相关）

两个基本点：最大投影反差、最小重构距离

K-L变换分连续和离散两种情况，这里只讨论**离散K-L变换法**。

1. K-L展开式

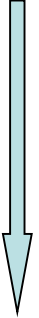
设 \mathbf{X} 是 n 维随机模式向量, 可以用完备归一化正交向量系 $\{\mathbf{u}_j\}$ 中的正交向量**展开式**:

$$\mathbf{X} = \sum_{j=1}^n a_j \mathbf{u}_j \quad a_j: \text{随机系数};$$

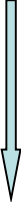
用有限项 ($d < n$) 估计 \mathbf{X} 时: $\hat{\mathbf{X}} = \sum_{j=1}^d a_j \mathbf{u}_j$

引起的均方误差: $\xi = E[(\mathbf{X} - \hat{\mathbf{X}})^T (\mathbf{X} - \hat{\mathbf{X}})]$

代入 \mathbf{X} 、 $\hat{\mathbf{X}}$, 利用 $\mathbf{u}_i^T \mathbf{u}_j = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}$


$$\xi = E\left[\sum_{j=d+1}^n a_j^2\right]$$

$$\xi = E\left[\sum_{j=d+1}^n a_j^2\right]$$

由 $\mathbf{X} = \sum_{j=1}^n a_j \mathbf{u}_j$ 两边  左乘 \mathbf{u}_j^T 得 $a_j = \mathbf{u}_j^T \mathbf{X}$ 。

$$\xi = E\left[\sum_{j=d+1}^n \mathbf{u}_j^T \mathbf{X} \mathbf{X}^T \mathbf{u}_j\right]$$

$$= \sum_{j=d+1}^n \mathbf{u}_j^T E[\mathbf{X} \mathbf{X}^T] \mathbf{u}_j$$

\mathbf{u}_j 为确定性向量

$$= \sum_{j=d+1}^n \mathbf{u}_j^T \mathbf{R} \mathbf{u}_j \quad \mathbf{R}: \text{自相关矩阵。}$$

不同的 $\{\mathbf{u}_j\}$ 对应不同的均方误差， \mathbf{u}_j 的选择应使 ξ 最小。

利用拉格朗日乘数法求使 ξ 最小的正交系 $\{\mathbf{u}_j\}$ ，令

$$g(\mathbf{u}_j) = \sum_{j=d+1}^n \mathbf{u}_j^T \mathbf{R} \mathbf{u}_j - \sum_{j=d+1}^n \lambda_j (\mathbf{u}_j^T \mathbf{u}_j - 1) \quad \lambda_j: \text{拉格朗日乘数}$$

$$g(\mathbf{u}_j) = \sum_{j=d+1}^n \mathbf{u}_j^T \mathbf{R} \mathbf{u}_j - \sum_{j=d+1}^n \lambda_j (\mathbf{u}_j^T \mathbf{u}_j - 1)$$

用函数 $g(\mathbf{u}_j)$ 对 \mathbf{u}_j 求导，并令导数为零，得

$$(\mathbf{R} - \lambda_j \mathbf{I}) \mathbf{u}_j = 0 \quad j = d + 1, \dots, n$$

——正是矩阵 \mathbf{R} 与其特征值和对应特征向量的关系式。

说明：当使用 \mathbf{X} 的自相关矩阵 \mathbf{R} 的特征值对应的特征向量来展开 \mathbf{X} 时，截断误差最小。

选前 d 项估计 \mathbf{X} 时引起的均方误差为

$$\xi = \sum_{j=d+1}^n \mathbf{u}_j^T \mathbf{R} \mathbf{u}_j = \sum_{j=d+1}^n \text{Tr}[\mathbf{u}_j \mathbf{R} \mathbf{u}_j^T] = \sum_{j=d+1}^n \lambda_j$$

λ_j 决定截断的均方误差， λ_j 的值小，那么 ξ 也小。

因此，当用 \mathbf{X} 的正交展开式中的前 d 项估计 \mathbf{X} 时，展开式中的 \mathbf{u}_j 应当是前 d 个较大的特征值对应的特征向量。

K-L变换具体方法:

对 R 的特征值由大到小进行排队: $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d \geq \lambda_{d+1} \geq \dots$

均方误差最小的 X 的近似式: $\mathbf{X} = \sum_{j=1}^d a_j \mathbf{u}_j$ —— K-L展开式

矩阵形式: $\mathbf{X} = \mathbf{U} \mathbf{a}$ (5-49)

式中, $\mathbf{a} = [a_1, a_2, \dots, a_d]^T$, $\mathbf{U}_{n \times d} = [\mathbf{u}_1, \dots, \mathbf{u}_j, \dots, \mathbf{u}_d]$ 。

其中: $\mathbf{u}_j = [u_{j1}, u_{j2}, \dots, u_{jn}]^T$

$$\mathbf{U}^T \mathbf{U} = \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \vdots \\ \mathbf{u}_d^T \end{bmatrix} [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \dots \quad \mathbf{u}_d] = \mathbf{I}$$

$$\mathbf{u}_i^T \mathbf{u}_j = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}$$

对式(5-49)两边左乘 \mathbf{U}^T : $\mathbf{a} = \mathbf{U}^T \mathbf{X}$ —— K-L变换式

系数向量 \mathbf{a} 就是变换后的模式向量—— X 主元值 (d 维),
而 $\{\mathbf{u}_j: j = 1, \dots, d\}$ 则是变换的 d 个主元方向 (n 维)。

2. 利用样本自相关矩阵进行K-L变换（特征提取）

设 \mathbf{X} 是 n 维模式向量， $\{\mathbf{X}\}$ 是来自 M 个模式类的样本集，总样本数目为 N 。将 \mathbf{X} 变换为 d 维 ($d < n$) 向量的方法：

第一步：求样本集 $\{\mathbf{X}\}$ 的总体自相关矩阵 \mathbf{R} 。

$$\mathbf{R} = E[\mathbf{X}\mathbf{X}^T] \approx \frac{1}{N} \sum_{j=1}^N \mathbf{X}_j \mathbf{X}_j^T$$

决定压缩
后的维数 d

第二步：求 \mathbf{R} 的特征值 λ_j ， $j = 1, 2, \dots, n$ 。对特征值由大到小进行排队，选择前 d 个较大的特征值。

第三步：计算 d 个特征值对应的特征向量 \mathbf{u}_j ， $j = 1, 2, \dots, d$ ，归一化后构成变换矩阵 \mathbf{U} 。

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_d]$$

第四步：对 $\{\mathbf{X}\}$ 中的每个 \mathbf{X} 进行 K-L 变换，得变换后向量 \mathbf{X}^* ：

$$\mathbf{X}^* = \mathbf{U}^T \mathbf{X}$$

d 维向量 \mathbf{X}^* 就是代替 n 维向量 \mathbf{X} 进行分类的模式向量。

例：两个二维模式类的样本分别为【注意：第三步的计算！】

$$\omega_1: \mathbf{X}_1 = [2, 2]^T, \mathbf{X}_2 = [2, 3]^T, \mathbf{X}_3 = [3, 3]^T$$

$$\omega_2: \mathbf{X}_4 = [-2, -2]^T, \mathbf{X}_5 = [-2, -3]^T, \mathbf{X}_6 = [-3, -3]^T$$

利用总体自相关矩阵 \mathbf{R} 作K-L变换，把原样本压缩成一维样本。

解：第一步：计算总体自相关矩阵 \mathbf{R} 。

$$\mathbf{R} = E\{\mathbf{X}\mathbf{X}^T\} = \frac{1}{6} \sum_{j=1}^6 \mathbf{X}_j \mathbf{X}_j^T = \begin{bmatrix} 5.7 & 6.3 \\ 6.3 & 7.3 \end{bmatrix}$$

第二步：计算 \mathbf{R} 的本征值，并选择较大者。由 $|\mathbf{R} - \lambda \mathbf{I}| = 0$ 得

$$\lambda_1 = 12.85, \lambda_2 = 0.15, \text{ 选择 } \lambda_1。$$

第三步：根据 $\mathbf{R}\mathbf{u}_1 = \lambda_1 \mathbf{u}_1$ 计算 λ_1 对应的特征向量 \mathbf{u}_1 ，令第1分量为1，归一化后为：

$$\mathbf{u}_1 = \frac{1}{\sqrt{2.3}} [1, 1.14]^T = [0.66, 0.75]^T$$

$$\mathbf{u}_1 = [0.66, 0.75]^T$$

变换矩阵为 $\mathbf{U} = [\mathbf{u}_1] = \begin{bmatrix} 0.66 \\ 0.75 \end{bmatrix}$

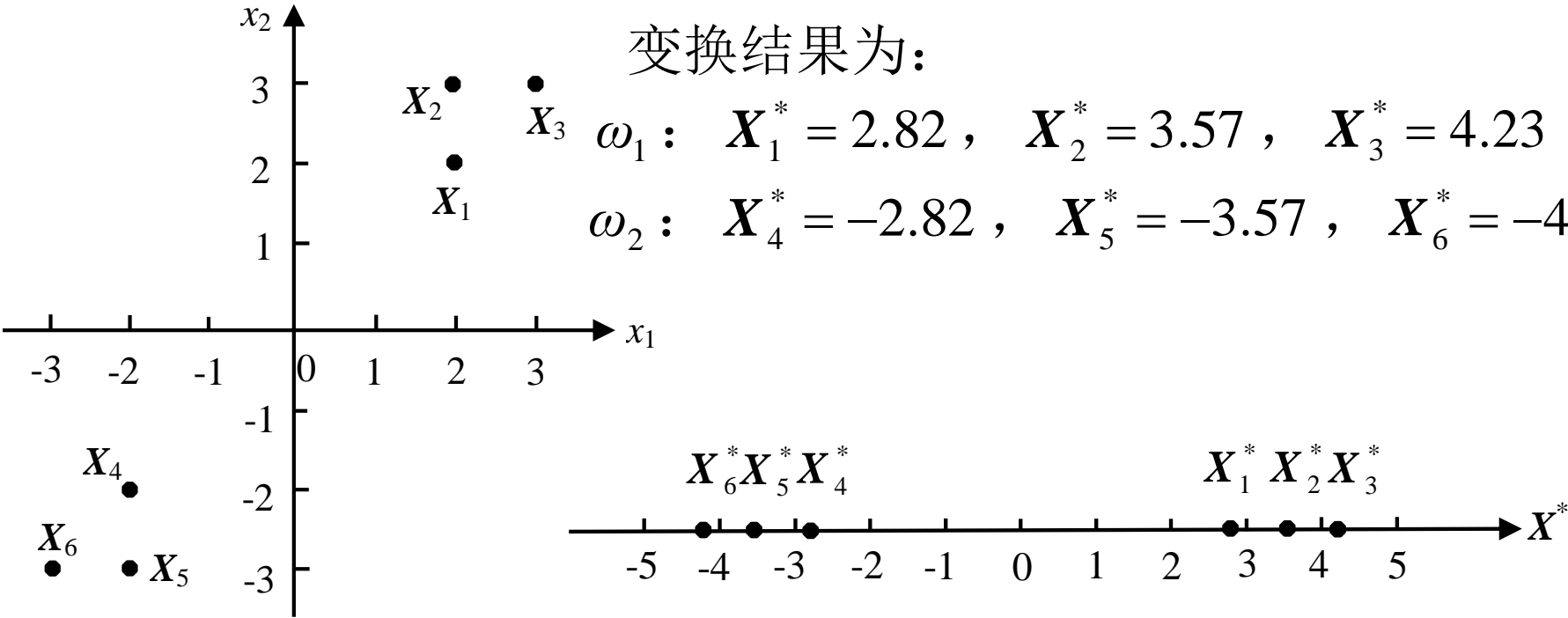
第四步：利用 \mathbf{U} 对样本集中每个样本进行 K-L 变换。

$$\mathbf{X}_1^* = \mathbf{U}^T \mathbf{X}_1 = [0.66 \ 0.75] \begin{bmatrix} 2 \\ 2 \end{bmatrix} = 2.82$$

.....

变换结果为：

$$\begin{aligned} \omega_1 : \mathbf{X}_1^* &= 2.82, \quad \mathbf{X}_2^* = 3.57, \quad \mathbf{X}_3^* = 4.23 \\ \omega_2 : \mathbf{X}_4^* &= -2.82, \quad \mathbf{X}_5^* = -3.57, \quad \mathbf{X}_6^* = -4.23 \end{aligned}$$



3. 使用不同散布矩阵进行K-L变换

根据不同的散布矩阵进行K-L变换，对保留分类鉴别信息的效果不同。

1) 采用多类类内散布矩阵 S_w 作 K-L 变换

多类类内散布矩阵：

$$S_w = \sum_{i=1}^c P(\omega_i) E[(X - M_i)(X - M_i)^T | X \in \omega_i]$$

若要突出各类模式的主要特征分量的分类作用：

选用对应于大特征值的特征向量组成变换矩阵；

若要使同一类模式聚集于最小的特征空间范围：

选用对应于小特征值的特征向量组成变换矩阵。

这一方法也是一种有监督的特征变换（降维）方法（见下一节）。

2) 采用类间散布矩阵 S_b 作 K-L 变换

$$\text{类间散布矩阵: } S_b = \sum_{i=1}^c P(\omega_i)(\mathbf{M}_i - \mathbf{M}_0)(\mathbf{M}_i - \mathbf{M}_0)^T$$

适用于类间距离比类内距离大得多的多类问题，选择与大特征值对应的特征向量组成变换矩阵。

3) 采用总体散布矩阵 S_t 作 K-L 变换

把多类模式合并起来看成一个总体分布。

$$\text{总体散布矩阵: } S_t = E[(\mathbf{X} - \mathbf{M}_0)(\mathbf{X} - \mathbf{M}_0)^T] = S_b + S_w$$

适合于多类模式在总体分布上具有良好的可分性的情况。

采用大特征值对应的特征向量组成变换矩阵，能够保留模式原有分布的主要结构。

利用K-L变换进行特征提取的优点:

- 1) 在均方逼近误差最小的意义下使新样本集 $\{\mathbf{X}^*\}$ 逼近原样本集 $\{\mathbf{X}\}$ 的分布, 既压缩了维数、又保留了数据集的分布信息和类别鉴别信息。
- 2) 变换后的新模式向量各分量相对总体均值的方差等于原样本集总体自相关矩阵的大特征值, 表明变换加强了模式类之间的差异性。

$$\mathbf{C}^* = E\{(\mathbf{X}^* - \mathbf{M}^*)(\mathbf{X}^* - \mathbf{M}^*)^T\} = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_d \end{bmatrix}$$

- 3) \mathbf{C}^* 为对角矩阵说明了变换后样本各分量特征互不相关, 即消除了原分量特征间的相关性, 便于进一步进行特征的选择。

K-L变换的不足之处:

- 1) 对两类问题容易得到较满意的结果。**类别愈多，效果愈差。**
- 2) 需要通过足够多的样本估计样本集的协方差矩阵或其它类型的散布矩阵。**当样本数不足时，矩阵的估计会变得十分粗略**，变换的优越性也就不能充分地显示出来。
- 3) 矩阵的特征值和特征向量缺乏统一的快速算法，计算较困难。

四、基于类内散布矩阵的单类模式特征提取

特征提取的目的:

有监督降维

对某类模式: 压缩模式向量的维数。

对多类分类: 压缩模式向量的维数;

保留类别间的鉴别信息, 突出可分性。

特征提取方法:

若 $\{X \in \omega_i\}$ 是 ω_i 类的一个 n 维样本集, 将 X 压缩成 m 维

向量 X^* —— 寻找一个 $m \times n$ 矩阵 A , 并作变换:

$$\begin{array}{ccccc} & & X^* = AX & & \\ & \swarrow & | & \searrow & \\ m \times 1 & & m \times n & & n \times 1 \end{array} \quad (m < n)$$

注意: 维数降低后, 在新的 m 维空间里各模式类之间的分布规律应至少保持不变或更优化。

1. 根据类内散布矩阵确定变换矩阵

设 ω_i 类模式的均值向量为 \mathbf{M} ，类内散布矩阵(协方差矩阵)为 \mathbf{C} :

$$\mathbf{M} = E\{\mathbf{X}\}$$

$$\mathbf{C} = E\{(\mathbf{X} - \mathbf{M})(\mathbf{X} - \mathbf{M})^T\}$$

式中， \mathbf{X} 为 n 维向量， \mathbf{C} 为 $n \times n$ 的实对称矩阵。

设矩阵 \mathbf{C} 的 n 个特征值分别为 $\lambda_1, \lambda_2, \dots, \lambda_n$
任一特征值是满足

$$|\lambda \mathbf{I} - \mathbf{C}| = 0$$

的一个解。

假定 n 个特征值对应的 n 个特征向量为 \mathbf{u}_k , $k = 1, 2 \cdots n$ 。

则 \mathbf{u}_k 是满足

$$\mathbf{C}\mathbf{u}_k = \lambda_k \mathbf{u}_k$$

的一个非零解。

\mathbf{u}_k 是 n 维向量, 可表示为 $[\mathbf{u}_{k1}, \mathbf{u}_{k2}, \cdots, \mathbf{u}_{kn}]$

若 \mathbf{u}_k 为归一化特征向量, 根据实对称矩阵的性质, 有

$$\mathbf{u}_i^T \mathbf{u}_j = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}$$

—— n 个特征向量相互正交。

若选 n 个归一化特征向量作为 $(\mathbf{A})_{n \times n}$ 的行, 则 \mathbf{A} 为归一化正交矩阵:

$$\mathbf{A} = \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \cdots \\ \mathbf{u}_n^T \end{bmatrix}$$

$$\mathbf{A}^T = [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \cdots \quad \mathbf{u}_n]$$

$$\mathbf{A}\mathbf{A}^T = \mathbf{I}$$

注意: 这里的 \mathbf{A} 对应于前一节的 \mathbf{U}^T

利用 \mathbf{A} 对 ω_i 类的样本 \mathbf{X} 进行变换, 得 $\mathbf{X}^* = \mathbf{A}\mathbf{X}$ 。

式中, \mathbf{X} 和 \mathbf{X}^* 都是 n 维向量。

$\mathbf{A}_{n \times n}$

考察变换前后的分布规律:

均值向量 \mathbf{M}^* 、协方差矩阵 \mathbf{C}^* 和类内距离 $\overline{D^2}$ 的变化。

$$(1) \mathbf{M}^* = E\{\mathbf{X}^*\} = E\{\mathbf{A}\mathbf{X}\} = \mathbf{A}E\{\mathbf{X}\} = \mathbf{A}\mathbf{M}$$

$$(2) \mathbf{C}^* = E\{(\mathbf{X}^* - \mathbf{M}^*)(\mathbf{X}^* - \mathbf{M}^*)^T\} = E\{(\mathbf{A}\mathbf{X} - \mathbf{A}\mathbf{M})(\mathbf{A}\mathbf{X} - \mathbf{A}\mathbf{M})^T\} \\ = \mathbf{A}E\{(\mathbf{X} - \mathbf{M})(\mathbf{X} - \mathbf{M})^T\}\mathbf{A}^T = \mathbf{A}\mathbf{C}\mathbf{A}^T$$

$$= \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \dots \\ \mathbf{u}_n^T \end{bmatrix} \mathbf{C} [\mathbf{u}_1 \ \mathbf{u}_2 \ \dots \ \mathbf{u}_n] = \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \dots \\ \mathbf{u}_n^T \end{bmatrix} [\lambda_1 \mathbf{u}_1 \ \lambda_2 \mathbf{u}_2 \ \dots \ \lambda_n \mathbf{u}_n] = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_n \end{bmatrix}$$

$$\mathbf{C}\mathbf{u}_k = \lambda_k \mathbf{u}_k$$

$$\mathbf{u}_i^T \mathbf{u}_j = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}$$

变换后：

协方差矩阵为对角阵，说明 \mathbf{X}^* 的各分量线性无关！

——便于特征的取舍；

\mathbf{X}^* 的第 k 个分量的方差等于未变换时 \mathbf{C} 的特征值 λ_k 。

(3) 变换后的类内均方距离

$$\begin{aligned}\overline{D^2} &= E\{\|\mathbf{X}_i^* - \mathbf{X}_j^*\|^2\} \\&= E\{(\mathbf{X}_i^* - \mathbf{X}_j^*)^T (\mathbf{X}_i^* - \mathbf{X}_j^*)\} \\&= E\{(\mathbf{A}\mathbf{X}_i - \mathbf{A}\mathbf{X}_j)^T (\mathbf{A}\mathbf{X}_i - \mathbf{A}\mathbf{X}_j)\} \\&= E\{(\mathbf{X}_i - \mathbf{X}_j)^T \mathbf{A}^T \mathbf{A} (\mathbf{X}_i - \mathbf{X}_j)\} \\&= E\{(\mathbf{X}_i - \mathbf{X}_j)^T (\mathbf{X}_i - \mathbf{X}_j)\} \\&= E\{\|\mathbf{X}_i - \mathbf{X}_j\|^2\}\end{aligned}$$

$$\mathbf{C}^* = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_n \end{bmatrix}$$

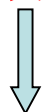
变换后的类内距离保持不变。注意 \mathbf{A} 是 $n \times n$ ，不是前面的 $m \times n$ 。

根据以上特点得到构造变换矩阵的方法：

目标：构造一变换矩阵，可以将 n 维向量 \mathbf{X} 变换成 m 维（ $m < n$ ）。

思路：

将变换前的 \mathbf{C} 的 n 个特征值从小到大排队



选择前 m 个小的特征值对应的特征向量
作为矩阵 \mathbf{A} 的行（ $m \times n$ ）

后↓续

对 \mathbf{X} 进行 \mathbf{A} 变换

$$\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ \mathbf{K} \\ x_n \end{bmatrix} \Rightarrow \begin{bmatrix} x_1^* \\ \vdots \\ x_m^* \end{bmatrix}$$

优点：压缩了维数；

类内距离减小，样本更密集——适合分类，不适合逼近！
——相当去掉了方差大的特征分量。

2. 针对样本集的特征提取具体步骤

设 $\{\mathbf{X}\}$ 为 ω_i 类的样本集， \mathbf{X} 为 n 维向量。

第一步：根据样本集求 ω_i 类的协方差矩阵（类内散布矩阵）。

$$\mathbf{C} = \frac{1}{N} \sum_{i=1}^N (\mathbf{X}_i - \mathbf{M})(\mathbf{X}_i - \mathbf{M})^T$$

其中，

$$\mathbf{M} = \frac{1}{N} \sum_{i=1}^N \mathbf{X}_i$$

第二步：计算 \mathbf{C} 的特征值，对特征值从小到大进行排队，选择前 m 个。

第三步：计算前 m 个特征值对应的特征向量 $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_m$ ，并归一化处理。将归一化后的特征向量的转置作为矩阵 \mathbf{A} 的行。

$$\mathbf{A} = \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \vdots \\ \mathbf{u}_m^T \end{bmatrix}$$

第四步：利用 \mathbf{A} 对样本集 $\{\mathbf{X}\}$ 进行变换。

$$\mathbf{X}^* = \mathbf{A}\mathbf{X}$$

则 m 维（ $m < n$ ）模式向量 \mathbf{X}^* 就是作为分类用的模式向量。

例 假定 ω_i 类的样本集为 $\{\mathbf{X}\} = \{\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3\}$ ，三个样本分别为

$$\mathbf{X}_1 = [1, 1]^T, \quad \mathbf{X}_2 = [2, 2]^T, \quad \mathbf{X}_3 = [3, 1]^T$$

用类内散布矩阵进行特征提取，将二维样本变换成一维样本。

解：1) 求样本均值向量和协方差矩阵。

$$\mathbf{M} = \frac{1}{3} \sum_{i=1}^3 \mathbf{X}_i = [2, 1.3]^T$$

$$\mathbf{C} = \frac{1}{3} \sum_{i=1}^3 \mathbf{X}_i \mathbf{X}_i^T - \mathbf{M} \mathbf{M}^T = \begin{bmatrix} 0.7 & 0.1 \\ 0.1 & 0.3 \end{bmatrix}$$

第二步：根据 $|\lambda \mathbf{I} - \mathbf{C}| = 0$ 求 \mathbf{C} 的特征值，并进行选择。

由
$$\begin{vmatrix} \lambda - 0.7 & -0.1 \\ -0.1 & \lambda - 0.3 \end{vmatrix} = 0$$

$$\mathbf{C} = \begin{bmatrix} 0.7 & 0.1 \\ 0.1 & 0.3 \end{bmatrix}$$

得 $\lambda_1 = 0.2765 \quad \lambda_2 = 0.7236$

由 $\lambda_1 < \lambda_2 \quad \therefore$ 选 λ_1

第三步：计算 λ_1 对应的特征向量 \mathbf{u}_1 。由方程 $\mathbf{C}\mathbf{u}_1 = \lambda_1\mathbf{u}_1$ 得

$$\mathbf{u}_1 = [0.5, -2.1]^T$$

归一化处理有

$$\mathbf{u}_1 = \frac{1}{\sqrt{0.5^2 + 2.1^2}} [0.5, -2.1]^T = \frac{1}{\sqrt{4.66}} [0.5, -2.1]^T$$

由归一化特征向量 \mathbf{u}_1 构成变换矩阵 \mathbf{A} ：
$$\mathbf{A} = \frac{1}{\sqrt{4.66}} [0.5, -2.1]$$

第四步：利用 A 对 X_1, X_2, X_3 进行变换。

$$X_1^* = AX_1 = -0.74$$

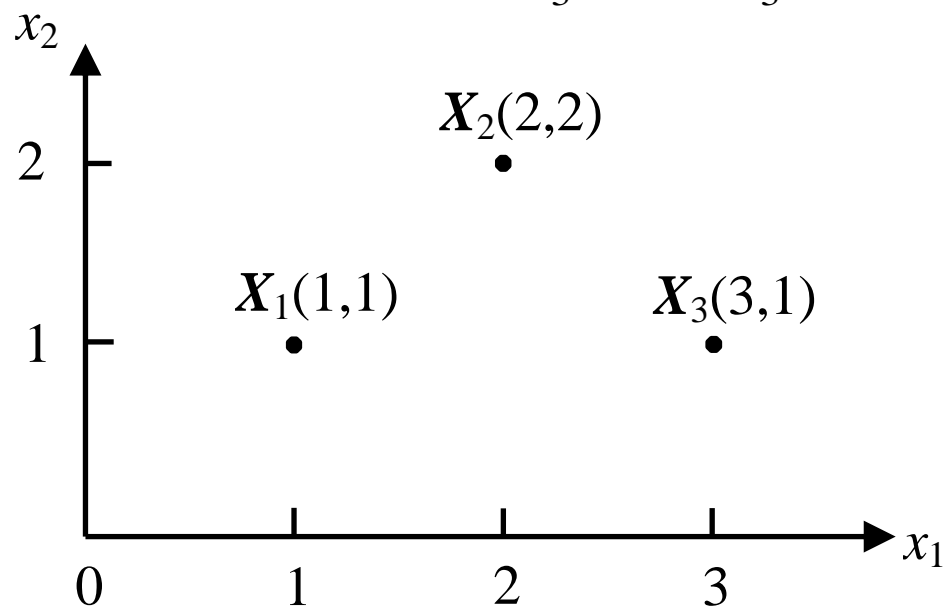
$$X_2^* = AX_2 = -1.48$$

$$X_3^* = AX_3 = -0.28$$

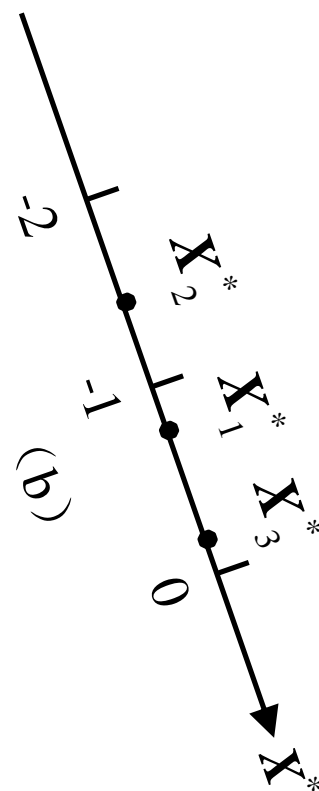
$$X_1 = [1, 1]^T$$

$$X_2 = [2, 2]^T$$

$$X_3 = [3, 1]^T$$



(a)
变换前

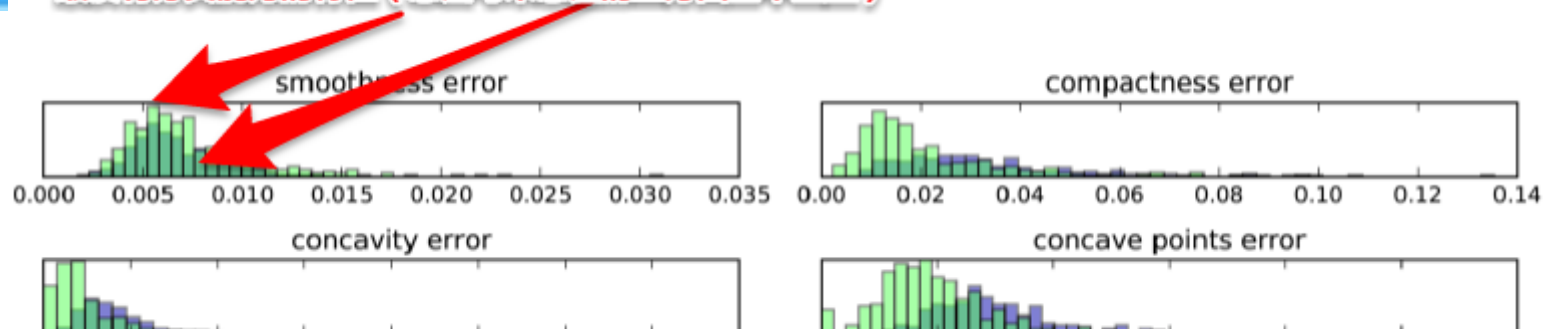


变换后

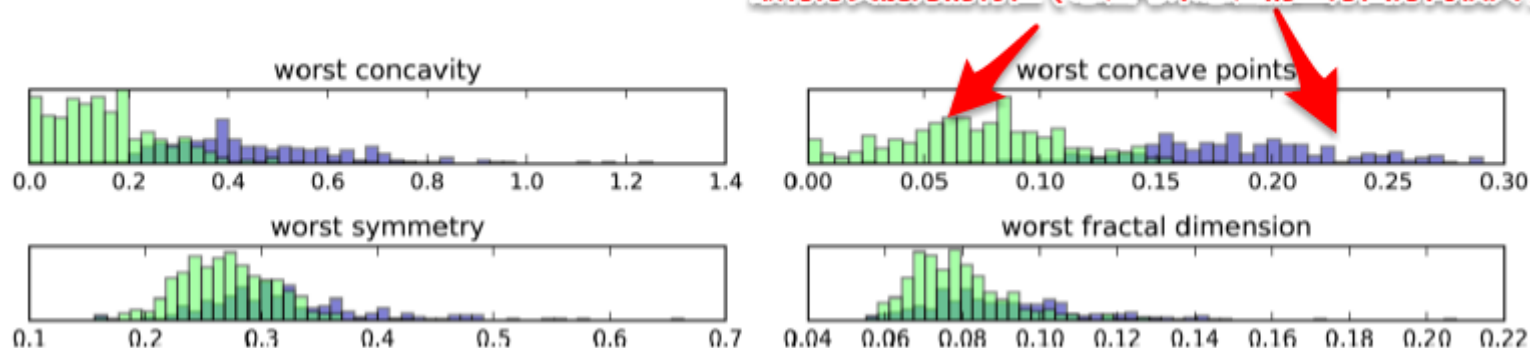
五、特征提取实验

示例1：乳腺癌数据的PCA分析（Muller, p.109）

最没有分类能力的特征（患癌与未患癌的直方图基本重叠）



最有分类能力的特征（患癌与未患癌的直方图分离较大）



In[14]:

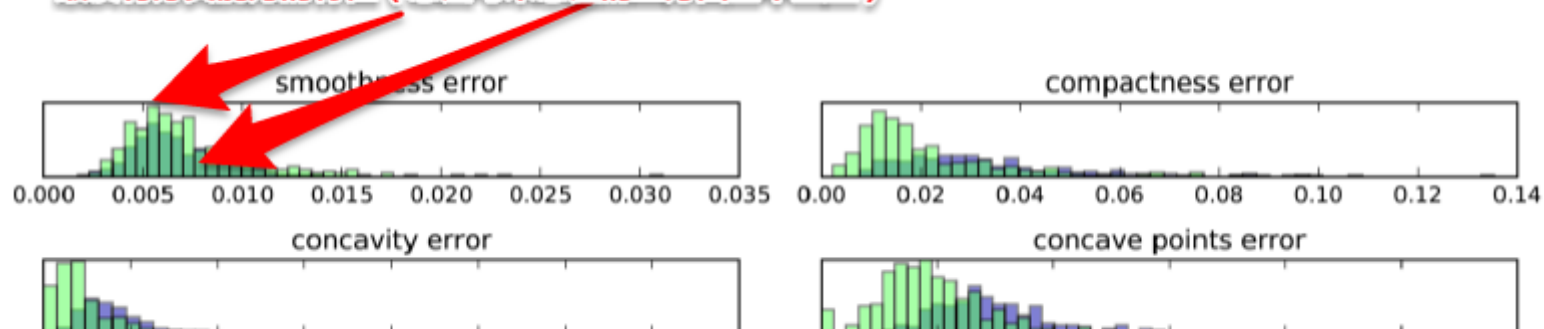
```
fig, axes = plt.subplots(15, 2, figsize=(10, 20))
malignant = cancer.data[cancer.target == 0]
benign = cancer.data[cancer.target == 1]

ax = axes.ravel()

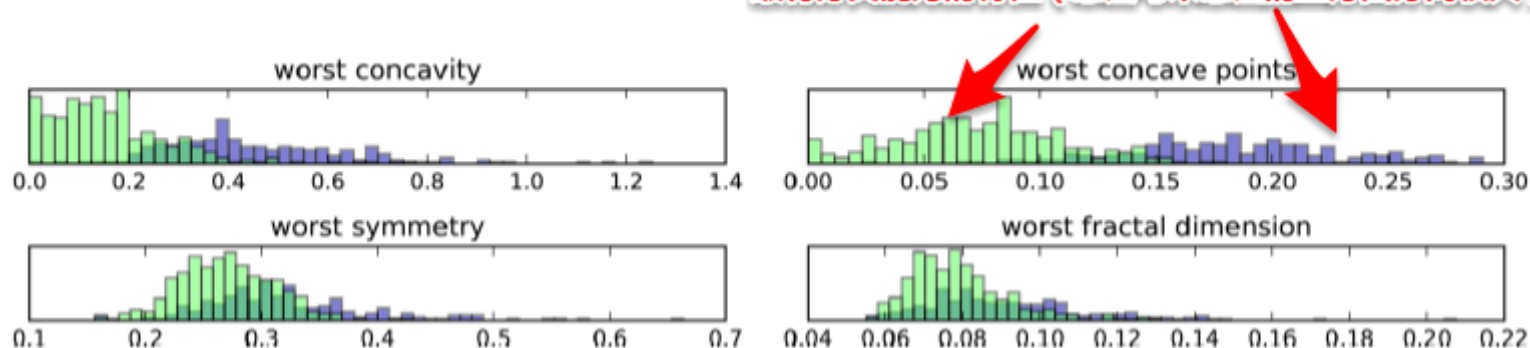
for i in range(30):
    _, bins = np.histogram(cancer.data[:, i], bins=50)
    ax[i].hist(malignant[:, i], bins=bins, color=mlearn.cm3(0), alpha=.5)
    ax[i].hist(benign[:, i], bins=bins, color=mlearn.cm3(2), alpha=.5)
    ax[i].set_title(cancer.feature_names[i])
    ax[i].set_yticks(())
ax[0].set_xlabel("Feature magnitude")
ax[0].set_ylabel("Frequency")
ax[0].legend(["malignant", "benign"], loc="best")
fig.tight_layout()
```

示例1：乳腺癌数据的PCA分析（Muller, p.109）

最没有分类能力的特征（患癌与未患癌的直方图基本重叠）



最有分类能力的特征（患癌与未患癌的直方图分离较大）

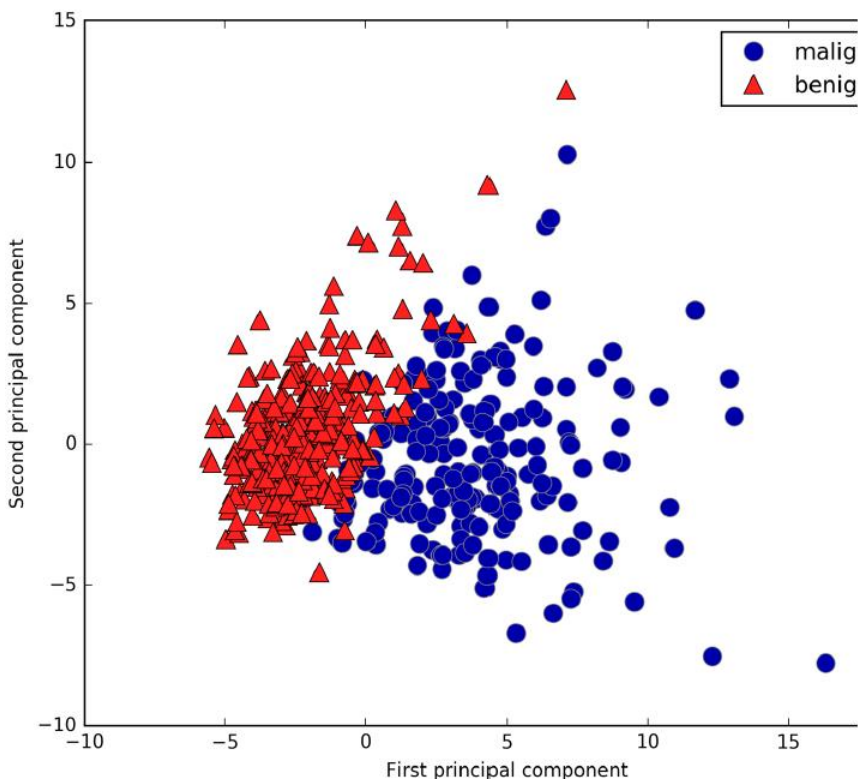


In[14]:

```
fig, axes = plt.subplots(15, 2, figsize=(10, 20))
malignant = cancer.data[cancer.target == 0]
benign = cancer.data[cancer.target == 1]

ax = axes.ravel()

for i in range(30):
    _, bins = np.histogram(cancer.data[:, i], bins=50)
    ax[i].hist(malignant[:, i], bins=bins, color=mlearn.cm3(0), alpha=.5)
    ax[i].hist(benign[:, i], bins=bins, color=mlearn.cm3(2), alpha=.5)
    ax[i].set_title(cancer.feature_names[i])
    ax[i].set_yticks(())
ax[0].set_xlabel("Feature magnitude")
ax[0].set_ylabel("Frequency")
ax[0].legend(["malignant", "benign"], loc="best")
fig.tight_layout()
```



每个样本只用两个主元
值表示，分类能力大

In[17]:

```
# plot first vs. second principal component, colored by class
plt.figure(figsize=(8, 8))
mglearn.discrete_scatter(X_pca[:, 0], X_pca[:, 1], cancer.target)
plt.legend(cancer.target_names, loc="best")
plt.gca().set_aspect("equal")
plt.xlabel("First principal component")
plt.ylabel("Second principal component")
```

In[15]:

```
from sklearn.datasets import load_breast_cancer
cancer = load_breast_cancer()
```

```
scaler = StandardScaler()
scaler.fit(cancer.data)
X_scaled = scaler.transform(cancer.data)
```

In[16]:

进行 2 维的 PCA

```
from sklearn.decomposition import PCA
# keep the first two principal components of the data
pca = PCA(n_components=2)
# fit PCA model to breast cancer data
pca.fit(X_scaled)

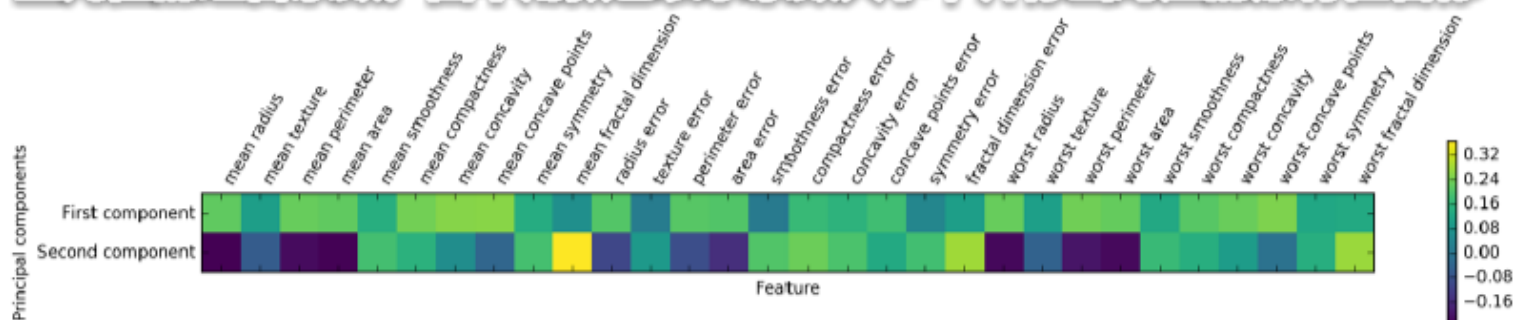
# transform data onto the first two principal components
X_pca = pca.transform(X_scaled)
print("Original shape: {}".format(str(X_scaled.shape)))
print("Reduced shape: {}".format(str(X_pca.shape)))
```

Out[16]: 569 个样本 (每个样本是 30 维特征)

Original shape: (569, 30)

Reduced shape: (569, 2)

热图：两个30维主元方向的每个分量值，其实就是作为计算各个样本的主元值的组合系数。图中用颜色表示系数大小，并标出对应的原特征名称



In[18]:

```
print("PCA component shape: {}".format(pca.components_.shape))
```

Out[18]:

这是根据数据集计算的两个主元方向

PCA component shape: (2, 30)



In[19]:

```
print("PCA components:\n{}".format(pca.components_))
```

Out[19]:

这是数据集的两个主元方向的具体值。各个样本的主元值就是通过原特征（30维）与两个主元方向分别内积得到。

PCA components:

```
[[ 0.219  0.104  0.228  0.221  0.143  0.239  0.258  0.261  0.138  0.064
  0.206  0.017  0.211  0.203  0.015  0.17  0.154  0.183  0.042  0.103
  0.228  0.104  0.237  0.225  0.128  0.21  0.229  0.251  0.123  0.132]
 [-0.234 -0.06 -0.215 -0.231  0.186  0.152  0.06 -0.035  0.19  0.367
 -0.106  0.09 -0.089 -0.152  0.204  0.233  0.197  0.13  0.184  0.28
 -0.22 -0.045 -0.2 -0.219  0.172  0.144  0.098 -0.008  0.142  0.275]]
```

示例2：人脸图像的PCA分析与低秩重建（Muller, p.113）



In[21]:

```
from sklearn.datasets import fetch_lfw_people
people = fetch_lfw_people(min_faces_per_person=20, resize=0.7)
image_shape = people.images[0].shape

fix, axes = plt.subplots(2, 5, figsize=(15, 8),
                        subplot_kw={'xticks': (), 'yticks': ()})
for target, image, ax in zip(people.target, people.images, axes.ravel()):
    ax.imshow(image)
    ax.set_title(people.target_names[target])
```

In[22]:

```
print("people.images.shape: {}".format(people.images.shape))  
print("Number of classes: {}".format(len(people.target_names)))
```

Out[22]:

每幅图片是 $87 \times 65 = 5655$ 维（原始特征），
涉及62人，3023幅图片（后进行采样封顶）

```
people.images.shape: (3023, 87, 65)  
Number of classes: 62
```

In[24]:

```
mask = np.zeros(people.target.shape, dtype=np.bool)  
for target in np.unique(people.target):  
    mask[np.where(people.target == target)[0][:50]] = 1
```

采样封顶：不超过50幅

```
X_people = people.data[mask]  
y_people = people.target[mask]
```

```
# scale the grayscale values to be between 0 and 1  
# instead of 0 and 255 for better numeric stability  
X_people = X_people / 255.
```


In[25]: 分层抽样，保证每个人的人脸样本都得到抽样。采用1-NN进行分类

```
from sklearn.neighbors import KNeighborsClassifier
# split the data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(
    X_people, y_people, stratify=y_people, random_state=0)
# build a KNeighborsClassifier using one neighbor
knn = KNeighborsClassifier(n_neighbors=1)
knn.fit(X_train, y_train)
print("Test set score of 1-nn: {:.2f}".format(knn.score(X_test, y_test)))
```

Out[25]:

Test set score of 1-nn: 0.27

mglearn.plots.plot_pca_whitening()

In[27]: 使用100维的PCA 白化，就是把各维特征标准化（变换为z-分值）。

```
pca = PCA(n_components=100, whiten=True, random_state=0).fit(X_train)
X_train_pca = pca.transform(X_train)
X_test_pca = pca.transform(X_test)

print("X_train_pca.shape: {}".format(X_train_pca.shape))
```

Out[27]: 3023幅图片，采样封顶+训练集-验证集分解，得到1537幅训练图片库。
100维的PCA后，得到如下的PCA 主元值（100维）

X_train_pca.shape: (1537, 100)

In[28]:

用最简单的1-NN分类

```
knn = KNeighborsClassifier(n_neighbors=1)  
knn.fit(X_train_pca, y_train)  
print("Test set accuracy: {:.2f}".format(knn.score(X_test_pca, y_test)))
```

Out[28]:

Test set accuracy: 0.36 准确率有所提高

In[29]:

```
print("pca.components_.shape: {}".format(pca.components_.shape))
```

Out[29]:

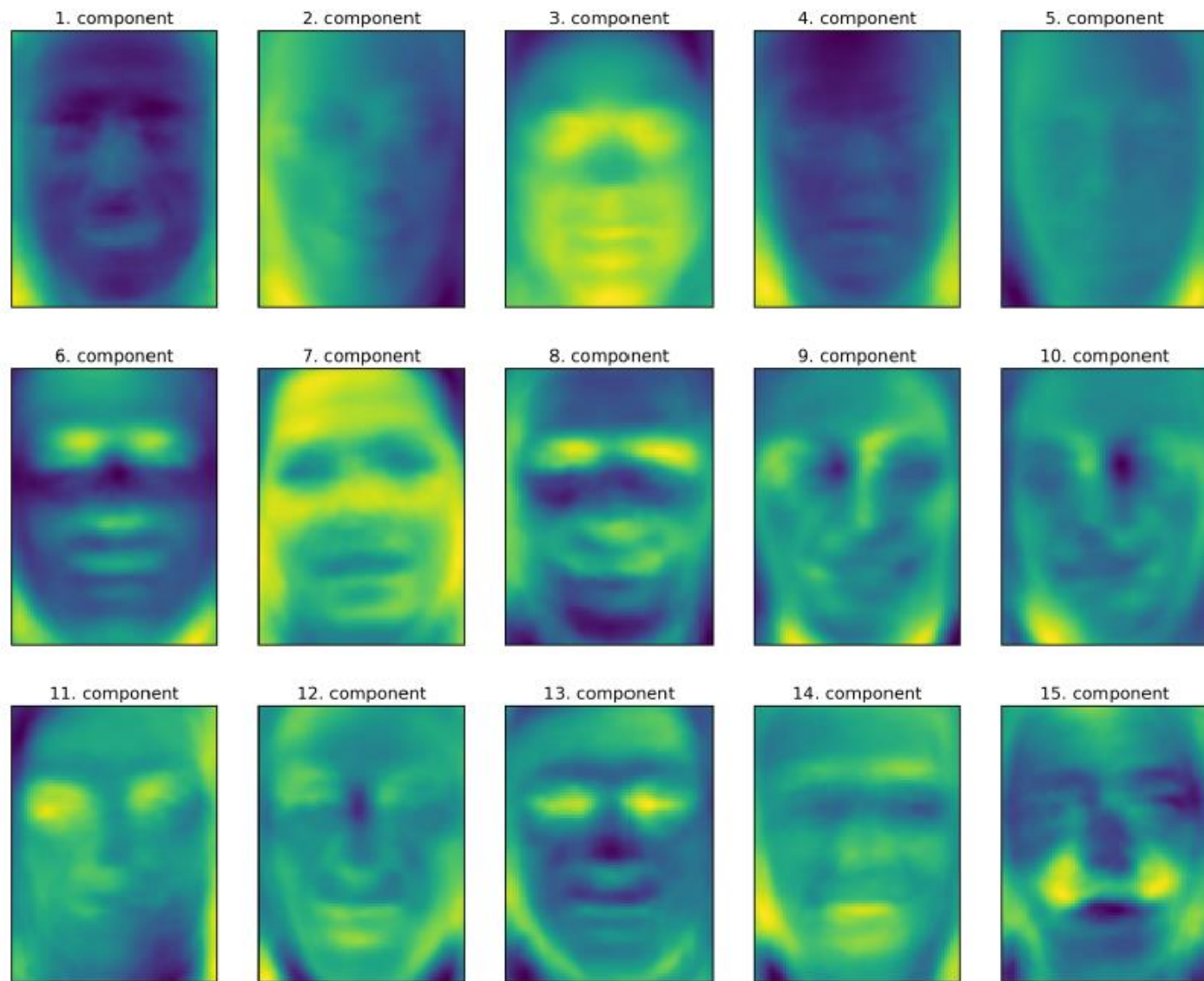
显示 进行PCA分析后的100维主元方向，把这些本无人脸像素意义的组合系数，强行转为“人脸”图像数据进行“显示”！

```
pca.components_.shape: (100, 5655)
```

In[30]:

用“人脸”模式，显示100个主元方向的前15个（3排，每排5幅）

```
fix, axes = plt.subplots(3, 5, figsize=(15, 12),  
                          subplot_kw={'xticks': (), 'yticks': ()})  
for i, (component, ax) in enumerate(zip(pca.components_, axes.ravel())):  
    ax.imshow(component.reshape(image_shape),  
             cmap='viridis')  
    ax.set_title("{} component".format(i + 1))
```

100维PCA主元方向的前15个主元方向的“图像”（特征脸）



使用各个样本的100个主元值，与100个主元方向组合，得到低秩重建的图像
 (低秩：指重建图像只使用了100个主元方向，而不是原特征数5655个主元方向)

$$\begin{array}{c} \text{original image} \end{array} \approx X_0 * \begin{array}{c} \text{feature map 1} \end{array} + X_1 * \begin{array}{c} \text{feature map 2} \end{array} + X_2 * \begin{array}{c} \text{feature map 3} \end{array} + X_3 * \begin{array}{c} \text{feature map 4} \end{array} + \dots$$

End of This Part