降维，特征选择

杨昆

计算机学院

杭州电子科技大学

降维

- 高维情形下出现的样本稀疏,距离计算困难等问题,是所有数据挖掘方法的共同障碍,被称为”维数灾难”(course of dimensionality)
- 一个重要的处理方法是降维(dimension reduction)-维数约简
  - 通过某种数学变换把高维属性空间转变成一个低维”子空间”,子空间里样本密度提高,距离计算也更容易.
- 降维为什么有效?
  - 虽然数据是高维,但与学习任务紧密相关的也许是某个低维分布(高维空间中的一个**低维”嵌入embedding”**).
- 效果评价-比较降维前后学习器的性能,若性能有所提高则认为降维有效果.

多维缩放MDS

◆ **经典降维方法**-Multiple Dimensional Scaling-MDS

- 要求原始空间中样本间的距离在低维空间中得以保持
- 假定m个样本在原空间的距离矩阵为$\mathrm{D} \in \mathbb{R}^{m\times m}$,第$i$行$j$列元素$dist_{ij}$为样本$x_i$与$x_j$的距离.

- 目标是获得样本在d'维空间的表示$\mathrm{Z} \in \mathbb{R}^{d'\times m}, d' \leq d$,且任意两个样本在$d'$维空间的欧氏距离等于原始空间中的距离,即$\|z_i - z_j\| = dist_{ij}$.

- 令降维后样本的内积矩阵为$\mathrm{B} = Z^T Z \in \mathbb{R}^{m\times m}$,则可以由降维前后保持不变的距离矩阵D计算内积矩阵B.

多维缩放MDS

◆ 算法

- 对B作特征值分解$\mathrm{B} = \mathrm{V}\Lambda V^{T}$,其中$\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_d)$是特征值构成的对角阵, V是特征向量矩阵,且令$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$.假设其中有$d^*$个非零特征值,它们构成对角矩阵$\Lambda_* = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_{d*})$,令相应的特征向量矩阵为$\mathrm{V}_*$,则Z可以表示成$\mathrm{Z} = \Lambda_*^{1/2} V_*^{T} \in \mathbb{R}^{d*\times m}$.

- 实际中,为了有效降维仅需降维前后的距离矩阵尽可能接近即可.此时取$d'(\ll d)$个最大特征值构成的对角阵$\widetilde{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_{d'})$和对应的特征向量矩阵$\tilde{V}$,则Z可以表示成$\mathrm{Z} = \widetilde{\Lambda}^{1/2} \tilde{V}^{T} \in \mathbb{R}^{d'\times m}$.

输入: 距离矩阵 $\mathbf{D} \in \mathbb{R}^{m \times m}$, 其元素 $dist_{ij}$ 为样本 $\boldsymbol{x}_i$ 到 $\boldsymbol{x}_j$ 的距离;
低维空间维数 d'.

过程:

1: 根据式(10.7)~(10.9)计算 $dist_{i\cdot}^2$, $dist_{\cdot j}^2$, $dist_{\cdot\cdot}^2$;
2: 根据式(10.10)计算矩阵 $\mathbf{B}$;
3: 对矩阵 $\mathbf{B}$ 做特征值分解;
4: 取 $\tilde{\mathbf{\Lambda}}$ 为 d' 个最大特征值所构成的对角矩阵, $\tilde{\mathbf{V}}$ 为相应的特征向量矩阵.

输出: 矩阵 $\tilde{\mathbf{V}}\tilde{\mathbf{\Lambda}}^{1/2} \in \mathbb{R}^{m \times d'}$, 每行是一个样本的低维坐标

$$dist_{i\cdot}^2 = \frac{1}{m}\sum_{j=1}^{m} dist_{ij}^2\,, (10.7)$$

$$dist_{\cdot j}^2 = \frac{1}{m}\sum_{i=1}^{m} dist_{ij}^2\,, (10.8)$$

$$dist_{\cdot\cdot}^2 = \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m} dist_{ij}^2\,, (10.9)$$

$$b_{ij} = -\frac{1}{2}\left(dist_{ij}^2 - dist_{i\cdot}^2 - dist_{\cdot j}^2 + dist_{\cdot\cdot}^2\right), (10.10)$$

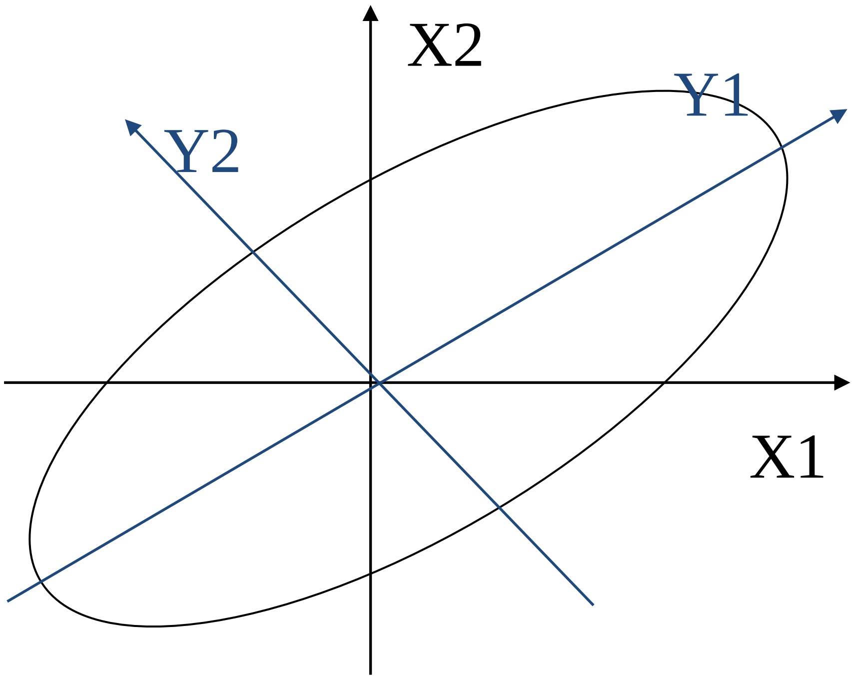
主成分分析

- 设两个变量X_1，X_2的N个样本在X_1和X_2的坐标空间中分布情况；无论沿X_1或X_2轴方向样本都有较大离散性（可以用方差表示），只考虑其中一个，原始数据的信息损失较大
- 考虑X_1和X_2的线性组合，使原始数据用新变量Y_1和Y_2表示
- 在几何上就是将坐标轴逆时针旋转θ角度，得到新坐标轴
- 旋转后N个样本在Y1轴上离散度最大，代表了原始数据的绝大部分信息。
- 目的：找到转换矩阵U

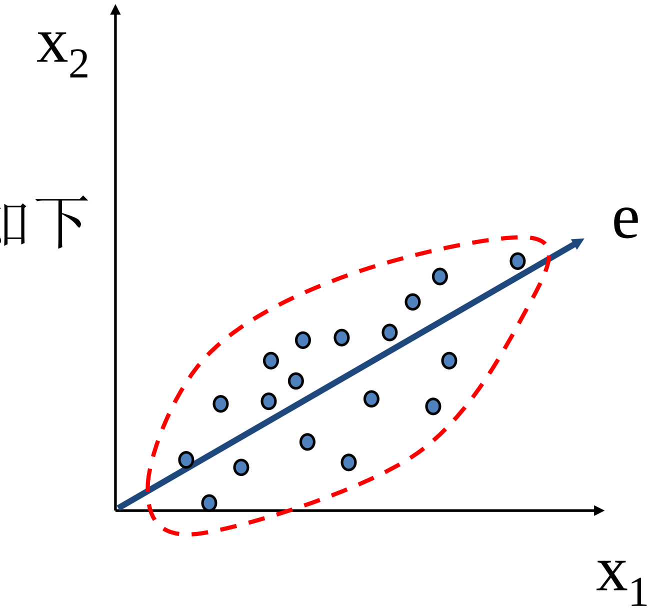
$$\begin{cases} Y_1 = \quad X_1 \cos\theta + X_2 \sin\theta \\ Y_2 = -X_1 \sin\theta + X_2 \cos\theta \end{cases}$$

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$$

Y=UX



主成分分析的基本思想



- 又称K-L变换，主成分与原始变量之间有如下基本关系：
  - 1)每个主成分都是原始变量的线性组合
  - 2)主成分的数目大大少于原始变量数
  - 3)主成分保留了原始变量绝大部分信息
  - 4)各主成分之间互不相关
- 对于变量$X_1, \ldots, X_p$，其协方差矩阵或相关矩阵就是各变量离散程度和变量间相关程度的反映。
- 实际求解主成分时，从原始数据的协方差矩阵或相关矩阵结构分析入手
- 求解主成分问题实际就是求特征根和特征向量问题
  - 何晓群. 多元统计分析(第三版),中国人民大学出版社，pp114-142

主成分分析的步骤

- ◆ **由协方差矩阵出发**
- ◆ **由相关矩阵出发**
  - ● 原始数据的相关矩阵实际上就是原始变量标准化后的协方差矩阵
- ◆ **两者过程一致，但一般来说结果主成分有差别，有时候还很大。**
  - ● 对数据标准化的过程也是抹杀变量离散程度的差异的过程（标准化后变量方差相等于1），而变量的方法差异可能是数据的固有特点。---标准化要不要做？问题
- ◆ **从什么出发求解主成分，目前没有定论。**
  - ● 对同度量或取值范围在同量级的数据，直接从协方差矩阵求解为好

主成分求解主要步骤

- 1) 假设原始样本集中有m个样本,n个指标，则原始样本集可构成样本矩阵 X。 $X=(x_{ij})_{m\times n}$
- 2)样本矩阵标准化处理,得到各指标向量均值为0,方差为1的标准化矩阵 $z_{ij}=\frac{x_{ij}-\bar{x}_j}{\sigma_j}, i=1,2,\ldots,m;\ j=1,2,\ldots,n$
- 3)计算相关系数矩阵R,不妨设R=Z’*Z. $Z=(z_{ij})_{m\times n}$

$$r_{ij}=\frac{1}{m-1}\sum_{k=1}^{m}(z_{ki}-\overline{z_i})(z_{kj}-\overline{z_j}), i=1,2,\ldots,n;\ j=1,2,\ldots,n$$

- 4)计算R的n个特征值 $\lambda_1\ge\lambda_2\ge\ldots\ge\lambda_n$ ，计算其对应的单位特征向量 e_i,i=1,2,…n
- 5)前k个主成分所对应的k个n维单位特征向量e_i (i=1,2,…k)组成$n\times k$维矩阵Y。
- 6)计算标准化样本矩阵Z在k个特征向量上的投影，得到主成分样本矩阵F=$(f_{ij})_{m\times k}$

$$F=Z\times Y$$

主成分个数k的确定

◆主成分λ_k的方差贡献率 η_k及前k个的累积贡献率 β_k。

$$\eta_k = \frac{\lambda_k}{\sum_{i=1}^{n} \lambda_i} \qquad \beta_k = \frac{\sum_{j=1}^{k} \lambda_j}{\sum_{i=1}^{n} \lambda_i}$$

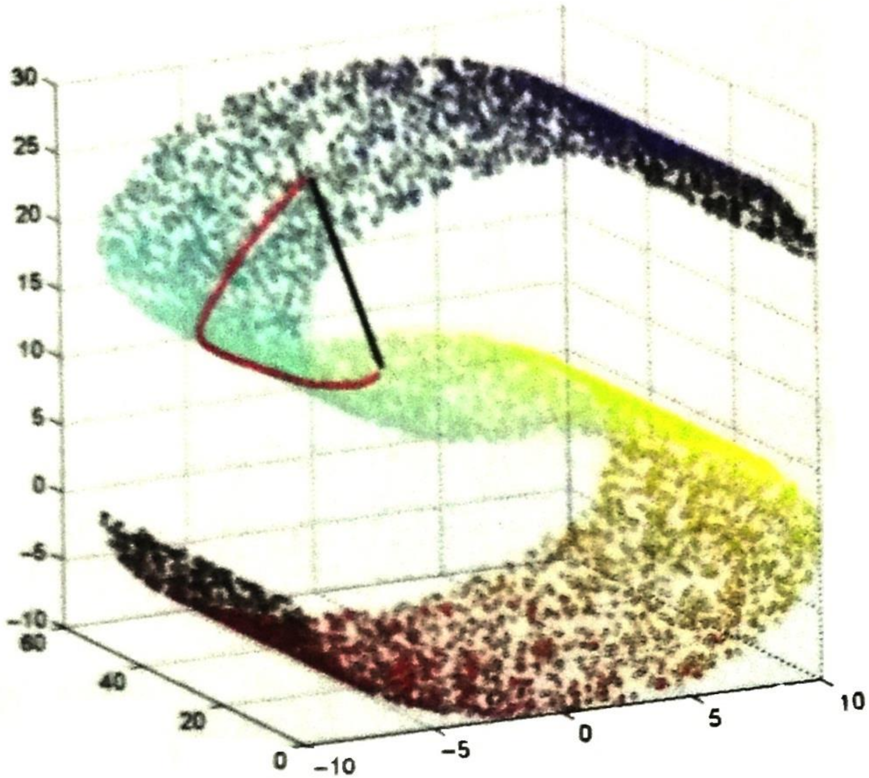
◆K取值多少合适？通常取k的值使得累积贡献率达到85%，90%为宜。

主成分分析的注意事项

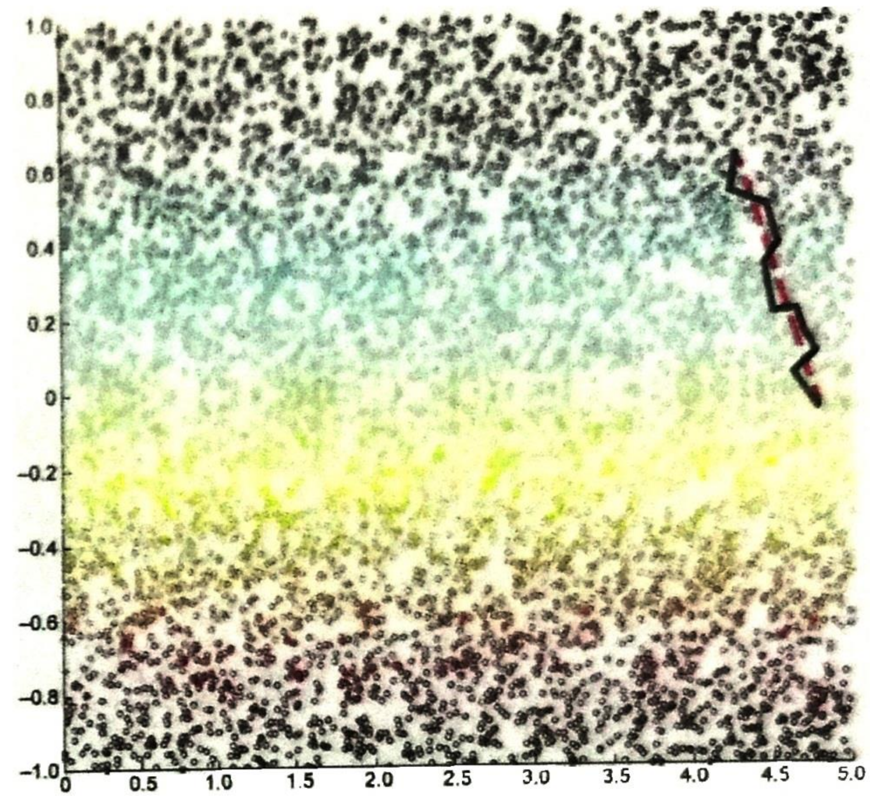
- **主成分分析不要求数据来自正态总体（主要用到矩阵分析技术）**
- **适用于当变量间存在较强相关性的数据，如果相关性弱则降维效果不好**
  - 一般认为，大部分变量间相关系数都小于0.3，不会有好效果
- **对重叠信息的剔出无能为力，同时还损失部分重要信息。**
  - 原始变量存在多重共线性时，应用主成分方法要慎重
- Works for numeric data only
- 不同的软件计算结果可能有差异。
  - 用相关系数矩阵R算出来特征值和特征向量,与用主成分函数princomp略微有点差异(matlab),实际上princomp调用奇异值分解来算特征向量.

等度量映射

- ◆ 等度量映射(Isometric Mapping-Isomap)
  - ●认为低维流行(manifold)嵌入到高维空间后,直接在高维



(a) 测地线距离与高维直线距离



(b) 测地线距离与近邻距离

等度量映射Isomap算法

输入: 样本集 $D = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m\}$;
近邻参数 k;
低维空间维数 d'.

过程:

1: **for** $i = 1, 2, \ldots, m$ **do**
2: 确定 $\boldsymbol{x}_i$ 的 k 近邻;
3: $\boldsymbol{x}_i$ 与 k 近邻点之间的距离设置为欧氏距离,
与其他点的距离设置为无穷大;
4: **end for**
5: 调用最短路径算法计算任意两样本点之间的距离 $\mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$;
6: 将 $\mathrm{dist}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ 作为 MDS 算法的输入;
7: **return** MDS 算法的输出

输出: 样本集 D 在低维空间的投影 $Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_m\}$.

- 权益之计:将训练样本的高维坐标为输入,低维坐标为输出,训练一个回归模型;然后对新样本的低维坐标进行预测.

等度量映射Isomap

◆ **近邻图的构建方法:**

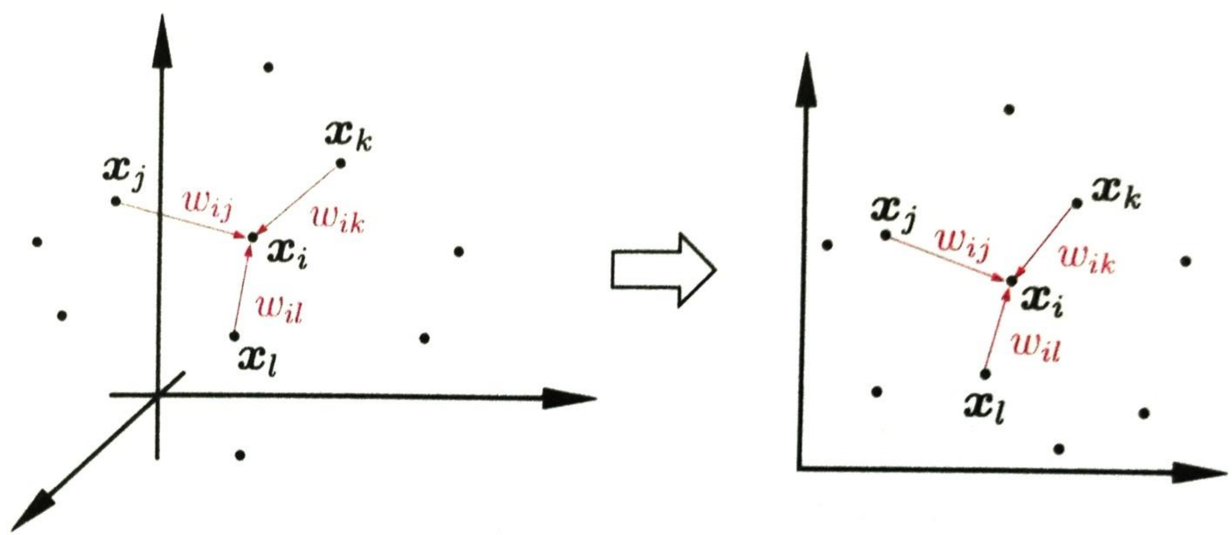
- (1)指定近邻点个数,这样得到的近邻图称为k近邻图
- (2)指定距离阈值ε,距离小于ε的点被认为是近邻点,得到的近邻称为近邻ε图.

◆ **两者兼有不足**

- "短路"问题:近邻范围指定得很大,则距离很远的点可能被误认为近邻;
- "断路"问题:近邻范围指定得很小,则图中有些区域和其他区域不存在连接;

局部线形嵌入

- 局部线形嵌入LLE-Locally Linear Embedding
  - 不同于Isomap,试图保持邻域内样本之间的线性关系
  - 假定样本x_i的坐标能通过它的邻域样本x_j, x_k, x_l的线性组合重构出来$x_i = w_{ij}x_j + w_{ik}x_k + w_{il}x_l$.LLE希望此关系在低维空间中得以保持.



局部线形嵌入

◆ 算法

- LLE先为每个样本x_i找到其近邻下标集合Q_i,然后计算出基于Q_i中的样本点对x_i进行线性重构的系数w_i:

$$\min_{w_1,w_2,\ldots,w_m} \sum_{i=1}^{m} \left\| x_i - \sum_{j\in Q_i} w_{ij} x_j \right\|_2^2, \quad (10.27)$$

$$s.t. \quad \sum_{j\in Q_i} w_{ij} = 1$$

- LLE在低维空间中保持w_i不变,于是x_i对应的低维空间坐标z_i可以通过下式求得

$$\min_{z_1,z_2,\ldots,z_m} \sum_{i=1}^{m} \left\| z_i - \sum_{j\in Q_i} w_{ij} z_j \right\|_2^2, \quad (10.28)$$

- 令$\mathrm{Z} = (z_1, z_2, \cdots, z_m) \in \mathbb{R}^{d\prime\times m}$, $(W)_{ij} = w_{ij}$,$\mathrm{M} = (I-W)^T(I-W)$,
- M的最小d’个特征值对应的特征向量组成的矩阵即为Z^T.

局部线形嵌入

◆LLE算法

输入: 样本集 $D=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_m\}$;
近邻参数 k; 低维空间维数 d'.

过程:

1: **for** $i=1,2,\ldots,m$ **do**
2: 　确定 $\boldsymbol{x}_i$ 的 k 近邻;
3: 　从式(10.27)求得 $w_{ij},\ j\in Q_i$;
4: 　对于 $j\notin Q_i$, 令 $w_{ij}=0$;
5: **end for**
6: 从式(10.30)得到 $\mathbf{M}$;
7: 对 $\mathbf{M}$ 进行特征值分解;
8: **return** $\mathbf{M}$ 的最小 d' 个特征值对应的特征向量

输出: 样本集 D 在低维空间的投影 $Z=\{\boldsymbol{z}_1,\boldsymbol{z}_2,\ldots,\boldsymbol{z}_m\}$.

$$\mathrm{M}=(I-W)^T(\mathrm{I}-W)\quad(10.30)$$

特征选择

- ◆ **特征选择**(feature selection)
  - 从给定的特征集合中选择出相关特征子集的过程(属于数据预处理中的一个重要环节)
  - 对当前任务有用的特征称为“相关特征relevant feature”
  - 没什么用的特征称为“无关特征irrelevant feature”
- ◆ **涉及两个关键环节,两者结合即得到特征选择方法**
  - “子集搜索subset search”问题:如何根据某原则(如评价结果)产生下一个候选子集?
    - ➢ “前向forward”搜索,“后向backward”搜索,“双向bidirectional”搜索
  - “子集评价subset evaluation”问题. 比如,信息增益等

特征选择的分类

- 过滤式filter
  - 过滤式方法先进行特征选择,再训练学习器,特征选择过程和后续学习器无关.---先用特征选择过程对初始特征进行“过滤”,再用过滤后特征训练模型.
- 包裹式wrapper
  - 直接把学习器的性能作为特征子集的评价标准.---为给定的学习器选择最有利其性能``量身定做”的特征子集.
  - 直接针对学习器优化,学习器性能 比过滤式更好
- 嵌入式embedding
  - (有别于前两者)特征选择过程和学习器训练过程融为一体,在同一个优化过程中完成,学习器训练过程中自动进行了特征选择

基于t检验统计量的方式

◆ 数学公式来说明

$$z=\frac{\overline{X}-\overline{Y}}{\sqrt{\frac{\sum_{i=1}^{n}(X_i-\overline{X})^2+\sum_{i=1}^{m}(Y_i-\overline{Y})^2}{n+m-2}\cdot\left(\frac{1}{n}+\frac{1}{m}\right)}}$$

两独立样本 *t* 检验—计算公式

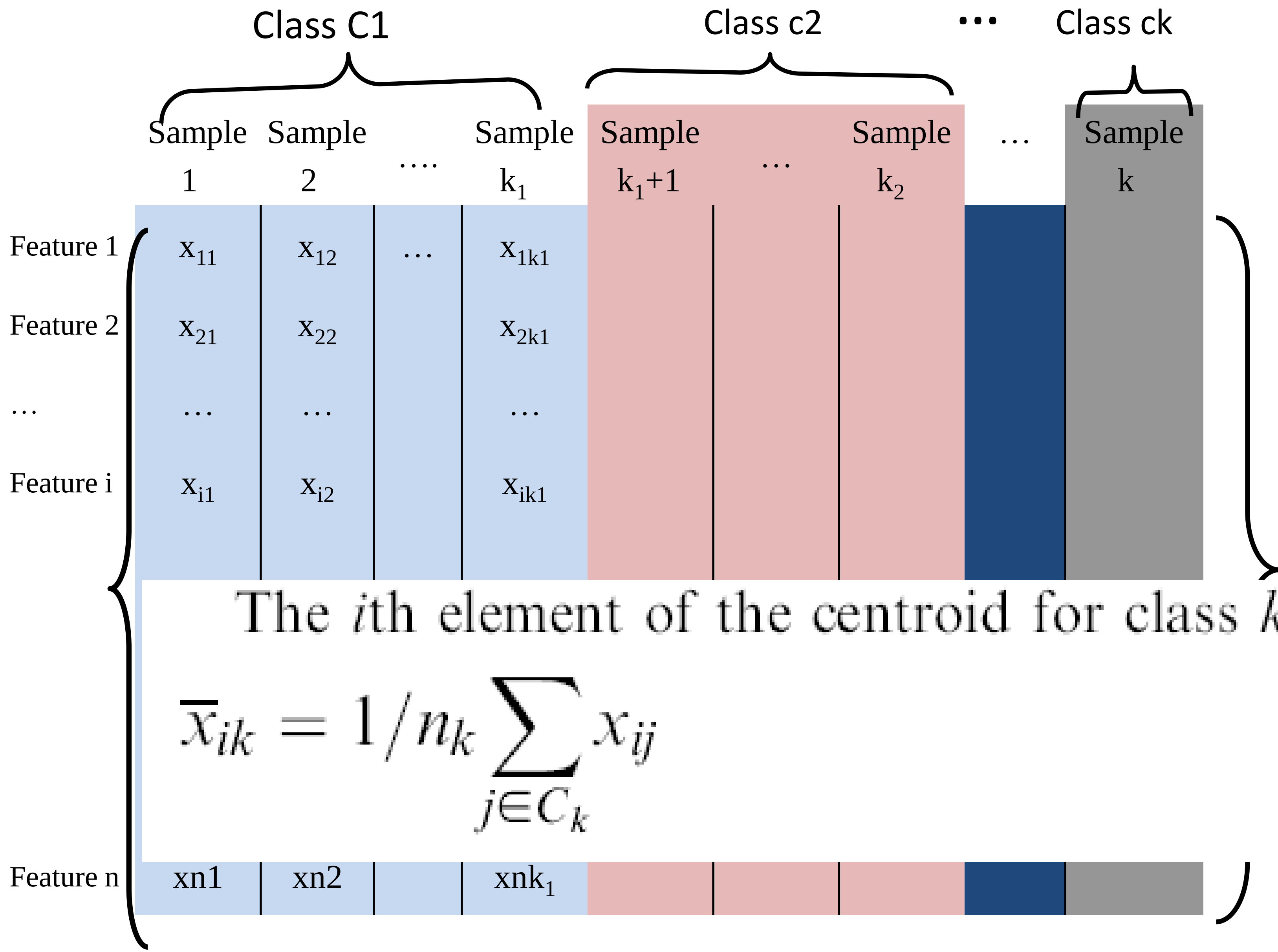
$$t=\frac{(\overline{X}_1-\overline{X}_2)-0}{S_{\overline{X}_1-\overline{X}_2}}=\frac{(\overline{X}_1-\overline{X}_2)}{S_{\overline{X}_1-\overline{X}_2}}$$

$$S_{\overline{X}_1-\overline{X}_2}=\sqrt{S_c^2\left(\frac{1}{n_1}+\frac{1}{n_2}\right)}$$ 为合并标准误

$$S_C{}^2=\frac{\sum X_1{}^2-\frac{(\sum X_1)^2}{n_1}+\sum X_2{}^2-\frac{(\sum X_2)^2}{n_2}}{n_1+n_2-2}$$

为称为合并方差，combined/pooled variance

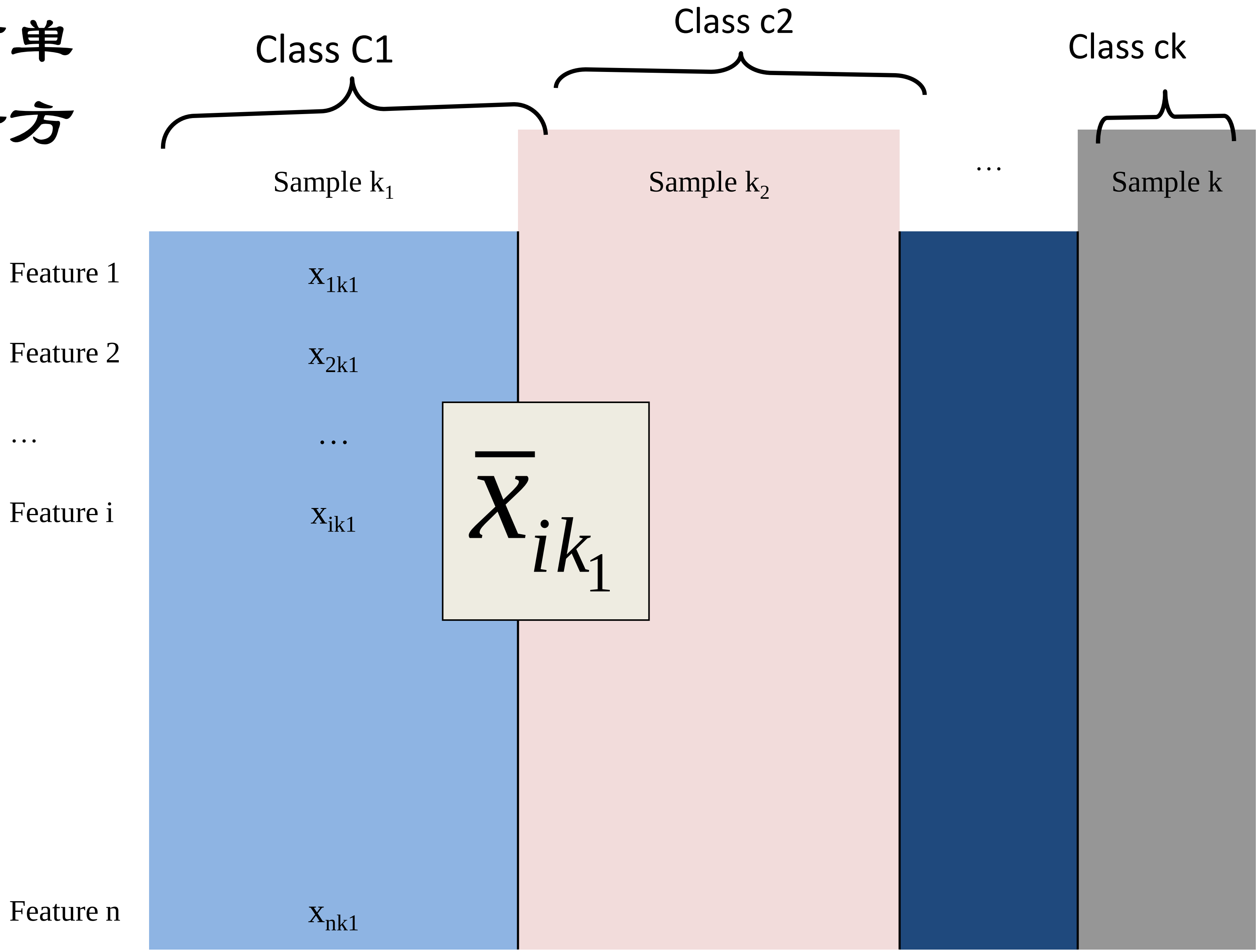
可以处理多类的简单计量方法



The *i*th element of the centroid for class *k*

$$\overline{x}_{ik} = 1/n_k \sum_{j \in C_k} x_{ij}$$

理多类的简单计量方法



Class C1
Class c2
Class ck
Sample k1
Sample k2
…
Sample k
Feature 1
Feature 2
…
Feature i
Feature n
x1k1
x2k1
…
xik1
xnk1
$\bar{x}_{ik_1}$

$$z_{ij} = \sqrt{(x_{ij}-\bar{x}_{ik})^2}, \text{ where } j \in C_k$$

Z_{ij}

	Sample 1	Sample 2	….	Sample k_1	Sample k_1+1	…	Sample k_2	…	Sample k
Feature 1	z11	z12	…	z1k_1					
Feature 2	z21	z22		z2k_1					
…	…	…		…					
Feature i	zi1	xi2		zik_1					
Feature n	zn1	zn2		znk_1					

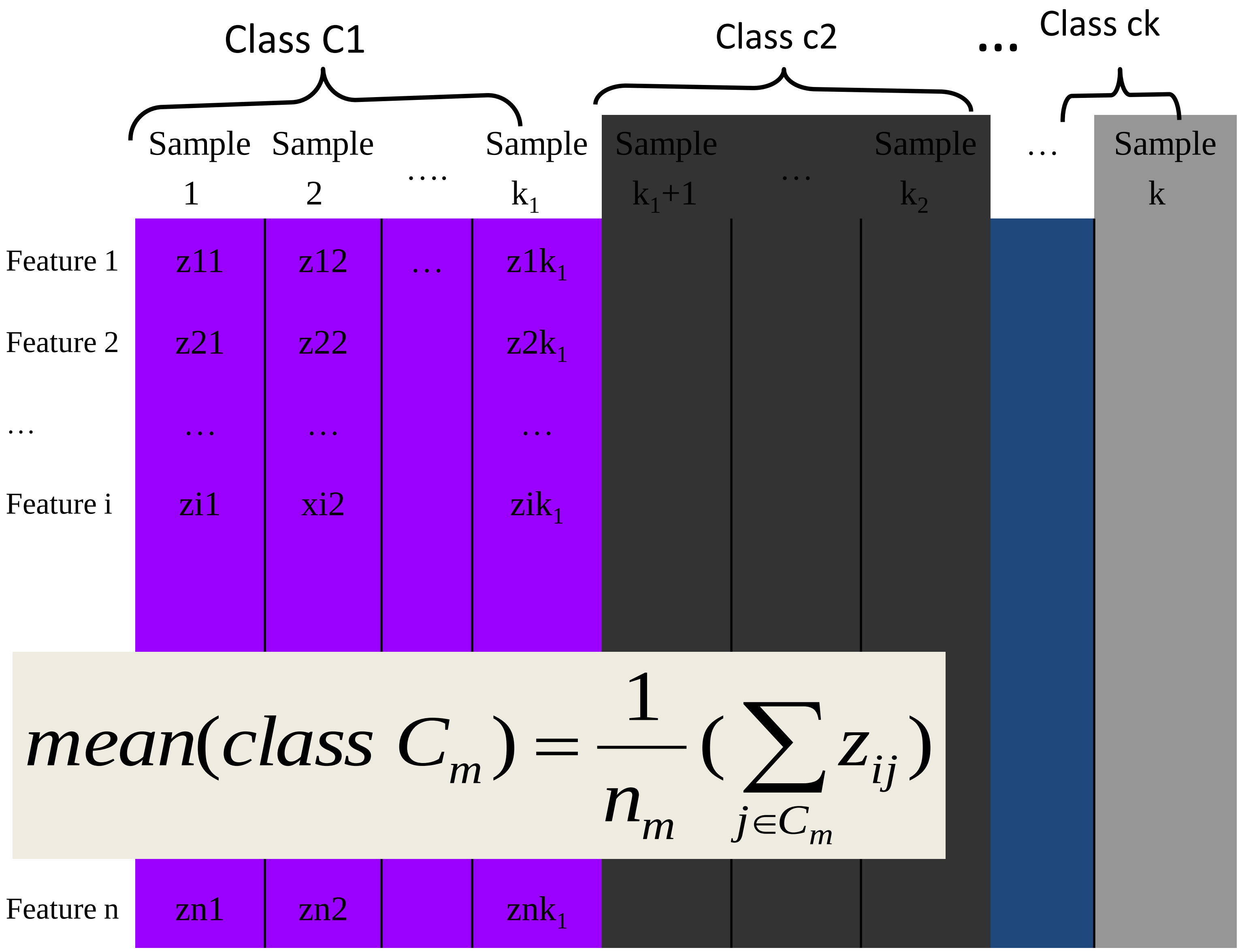
$$Z_i = \left(\sqrt{(X_{i1}-\bar{X}_{i1})^2}\ ,\ \sqrt{(X_{i2}-\bar{X}_{i1})^2}, \ldots, \sqrt{(X_{in_1}-\bar{X}_{i1})^2}, \cdots, \sqrt{(X_{in_2}-\bar{X}_{i2})^2}, \cdots\right)$$

Mean and Std of Feature i

$$\mathrm{mean}_w(\mathbf{z}_i) = \sum_{j=1}^{n} \frac{w_j}{W} z_{ij}$$

$$= \frac{\sum_{m=1}^{k} \left[\frac{1}{n_m} \left(\sum_{j \in C_m} z_{ij} \right) \right]}{k}$$

$$\mathrm{std}_w(\mathbf{z}_i) = \sqrt{\frac{\sum_{j=1}^{n} (z_{ij} - \mathrm{mean}_w(\mathbf{z}_i))^2}{(n-1/n) \sum_{j=1}^{n} w_j}}$$



$$mean(class\ C_m) = \frac{1}{n_m}\left(\sum_{j \in C_m} z_{ij}\right)$$

可以处理多类的简单计量方法

$$r_i = \mathrm{mean}_w(\mathbf{z}_i)\cdot\mathrm{std}_w(\mathbf{z}_i)$$

$$R_i = \frac{\mathrm{mean}_w(\mathbf{z}_i)\cdot\mathrm{std}_w(\mathbf{z}_i)}{\mathrm{std}(\overline{\mathbf{x}}_i)}$$

Ri small value表明第i个特征特征分散在每个类的质心附近，并且are assembled simultaneously(std)

可以处理多类的计量方法2

◆ BMC的一个方法

$$score(j)=\frac{compact(j)}{scatter(j)}=\frac{d_2(j)+\sqrt{d_2(j)^2-mean(\bar{X},j)^2}}{scatter(j)}$$

$$d_2(j)=\sqrt{\frac{1}{L}\sum_{k=1}^{L}\left(\frac{1}{n_k}\sum_{i\in C_k}x_{ij}^2\right)}$$

$$mean(\bar{X},j)=\frac{1}{L}\sum_{1}^{L}\bar{x}_{kj}=\bar{x}_j=\frac{1}{L}\sum_{k=1}^{L}\left(\frac{1}{n_k}\sum\nolimits_{i\in C_k}x_{ij}\right)$$

$$scatter(j)=\sqrt{\frac{1}{L}(\bar{a}_{kj}-\bar{a}_j)^2}+\frac{1}{2}\min_{w\neq v}\left|\bar{a}_{wj}-\bar{a}_{vj}\right|$$

Relief(Relevant Features)

- **设计一个“相关统计量”(向量)来度量特征的重要性**
  - 每个分量分别对应于一个初始特征,特征子集的重要性由子集中每个特征的对应统计量分量之和决定
  - (1)指定一个阈值τ,选择比τ大的统计量分量所对应的特征即可
  - (2)指定特征个数k,选取统计量分量最大的k个特征
- **(关键)如何计算相关统计量?**
  - 给定训练集$\{(x_1,y_1),\ldots,(x_m,y_m)\}$,对每个样本$x_i$,Relief先在同类中寻找最近邻$x_{i,nh}$,称为“猜中近邻”(near-hit),再从异类样本中找其最近邻$x_{i,nm}$,称为"猜错近邻"(near-miss),然后计算属性j的统计分量

$$\delta^j = \sum_i -diff\left(x_i^j, x_{i,nh}^j\right)^2 + diff\left(x_i^j, x_{i,nm}^j\right)^2$$

Relief

$$\delta^j = \sum_i -diff(x_i^j, x_{i,nh}^j)^2 + diff(x_i^j, x_{i,nm}^j)^2$$

- x_a^j表示样本x_a在属性j上的值,$diff(x_a^j, x_b^j)$取决于属性类型
  - 若离散型性则取1或0,连续型取$|x_a^j - x_b^j|$, x_a^j, x_b^j规范到[0,1]
- 属性j区分同类和异类样本是否有益?取决于x_i关于猜中近邻和猜错近邻在属性j上的距离比较
- 对不同样本上的结果进行平均,得到统计量分量,值越大则其属性的分类能力越强.
  - 实际上Relief只需在数据集的采样而不是整个集合上估计相关统计量
- 时间开销随采样数和原始特征数线形增加,高效算法

多分类的Relief-F

- Relief针对二分类问题设计;
- Relief-F能处理多类
  - 数据集D中样本来自$|Y|$个类.设样本x_i属于k类,则算法先在第k类中寻找x_i的最近邻$x_{i,nh}$为猜中近邻;
  - 然后在每个其他类中找一个x_i的最近邻为猜错近邻,记为$x_{i,l,nm}$,(l=1, ..., $|Y|$, $l \neq k$).
  - 令p_l为第l类样本在数据集D中的比例
  - 统计量对应于属性j的分量为

$$\delta^j = \sum_i -diff\left(x_i^j, x_{i,nh}^j\right)^2 + \sum_{l \neq k}\left(p_l \times diff\left(x_i^j, x_{i,l,nm}^j\right)^2\right)$$

包裹式选择

- LVW(Las Vegas Wrapper)在拉斯维加斯方法框架下使用随机策略来进行子集搜索,最终以分类器的误差为子集评价标准
- 每次特征子集评价都需要训练分类器,计算开销很大,因此设置了停止条件参数T

包裹式选择

- 第8行使用CV在特征子集A'上估计误差
- 比当前子集A的误差更小,或误差相当但A'中特征数目更少,则被保留

输入: 数据集 D; 特征集 A;

过程: 学习算法 $\mathfrak{L}$; 停止条件控制参数 T.

1: $E = \infty$;
2: $d = |A|$;
3: $A^* = A$;
4: $t = 0$;
5: **while** $t < T$ **do**
6: 　随机产生特征子集 A';
7: 　$d' = |A'|$;
8: 　$E' = \mathrm{CrossValidation}(\mathfrak{L}(D^{A'}))$;
9: 　**if** $(E' < E) \vee ((E' = E) \wedge (d' < d))$ **then**
10: 　　$t = 0$;
11: 　　$E = E'$;
12: 　　$d = d'$;
13: 　　$A^* = A'$
14: 　**else**
15: 　　$t = t + 1$
16: 　**end if**
17: **end while**

输出: 特征子集 A^*

启发式向前或向后的子集生成方法

基于SVM的缠绕方法

嵌入式选择

◆ **考虑简单的线性回归模型**

$$\min_{w}\sum_{i=1}^{m}(y_i - w^T x_i)^2$$

- 给定训练集$\{(x_1,y_1),\ldots,(x_m,y_m)\}$
- 当样本数少而特征多时,容易过拟合.引入**正则化项(参数λ>0)**.
- 若引入L_2范数正则化,称为"岭回归"ridge regression

$$\min_{w}\sum_{i=1}^{m}(y_i - w^T x_i)^2 + \lambda\|w\|^2{}_2$$

- 若采用L_1范数,则称为LASSO

$$\min_{w}\sum_{i=1}^{m}(y_i - w^T x_i)^2 + \lambda\|w\|_1$$

◆ **L_1范数有额外好处,所求的w会有更少的非零分量,即"稀疏解"**

嵌入式选择

- W是稀疏解意味着，初始d个特征中仅有w中非零分量对应的特征出现于最终模型中.
- 基于L_1正则化的方法是一种嵌入式特征选择方法
  - L_1正则化问题的解法可以使用近端梯度下降(Proximal Gradient Descent-PGD, 2005)

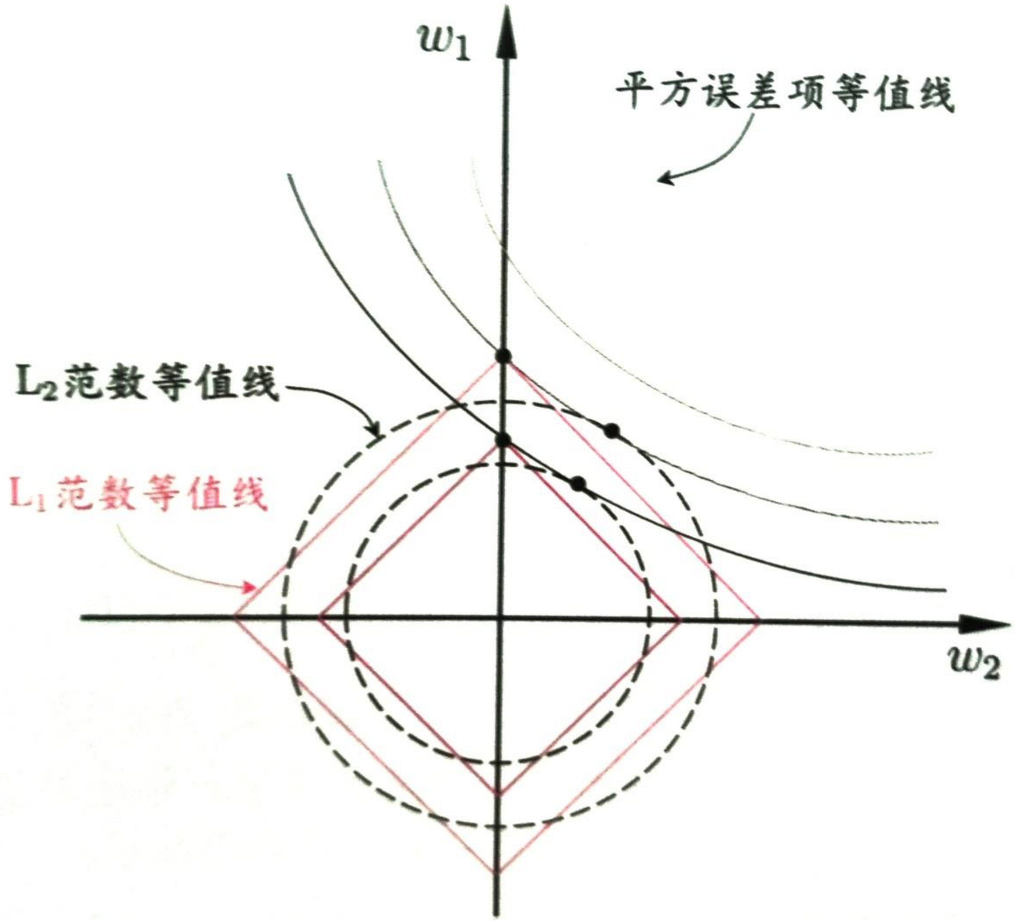


图 11.2 L_1 正则化比 L_2 正则化更易于得到稀疏解