机器学习

于元隆、朱丹红

福州大学计算机与大数据学院
Email: yu.yuanlong@fzu.edu.cn

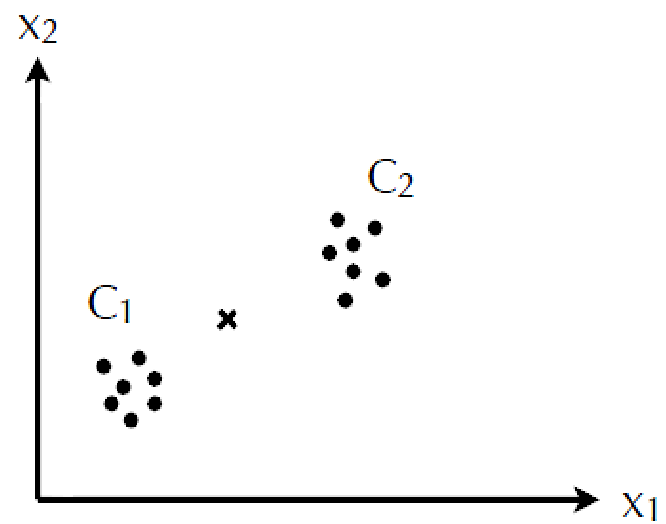
- 第二讲 基于距离的分类器

基于距离的分类

- 基于距离的分类：把测试样本到每个类之间的距离作为判断准则。该技术是最常见的模式识别技术，是其它高级识别技术的基础。
- 判别公式：

$$\mathbf{x} \in C_i, \text{ if } d(\mathbf{x}, C_i) \leq d(\mathbf{x}, C_j), \forall j \neq i$$

$\mathbf{x}$: Test pattern (sample)
C_i, C_j: Classes
d: Distance



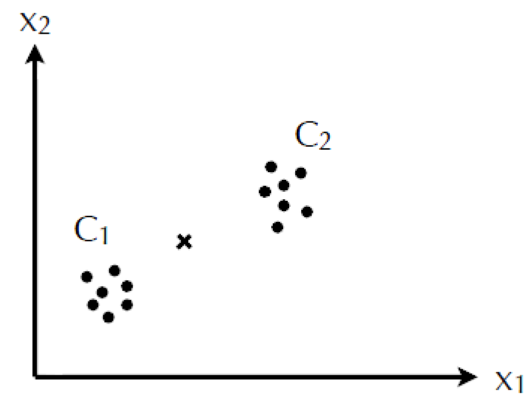
- 基于距离分类的两个关键因素：
  - 类的表达（原型）
  - 距离的衡量

类的原型

- 类的原型(prototype)：用来代表这个类的一个模式或者一组量，方便计算该原型和测试样本之间的距离。

$$d(\mathbf{x}, C_i) = d(\mathbf{x}, \mathbf{z}_i)$$

where $\mathbf{z}_i$ denotes the prototype of class i.



- 如何学习类的原型?
  - 统计值（参数化）
  - 概率密度估计（无参数）

- 如果使用类的原型，则基于距离的判别公式可以写为：

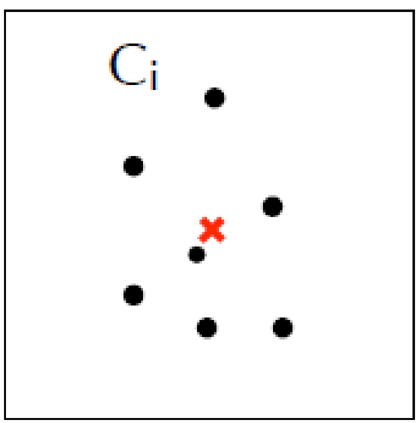
$$\mathbf{x} \in C_i, \text{ if } d(\mathbf{x}, \mathbf{z}_i) \leq d(\mathbf{x}, \mathbf{z}_j), \forall j \neq i$$

where $\mathbf{z}_i$ and $\mathbf{z}_j$ respectively denote the prototypes of class i and j.

类的原型：均值

- 原型的种类：
  - 均值：该类中所有训练样本的均值作为类的原型。
    - 一阶统计
    - 均值是对类中所有训练样本代表误差最小的一种表达方式

$$\mathbf{z}_i = \mathbf{m}_i = \frac{\sum_{\mathbf{x}\in C_i}\mathbf{x}}{N_i}$$



$$e_i = \sum\nolimits_{\boldsymbol{x}\in C_i} ||\boldsymbol{x}-\boldsymbol{z_i}||_2^2$$

$$= \sum\nolimits_{\boldsymbol{x}\in C_i} (\boldsymbol{x}-\boldsymbol{z_i})^T(\boldsymbol{x}-\boldsymbol{z_i})$$

为了最小化误差，令其梯度为0，即：

$$\frac{\partial err_i}{\partial \boldsymbol{z_i}} = \sum\nolimits_{\boldsymbol{x}\in C_i} (\boldsymbol{x}-\boldsymbol{z_i})$$

$$= \sum\nolimits_{\boldsymbol{x}\in C_i} \boldsymbol{x} - N_i\boldsymbol{z_i} = 0$$

$$\Rightarrow \boldsymbol{z}_i = \frac{\sum_{\boldsymbol{x}\in C_i}\boldsymbol{x}}{N_i}$$

类的原型：概率密度估计

无参数概率密度估计的任务描述

- 在只给定N个训练样本，不知道概率分布形式的条件下，在特征空间内学习每个任意取值（模式）$\boldsymbol{x}$的概率密度；
- 也就是估计以$\boldsymbol{x}$为中心、在极小的区域$R = (\boldsymbol{x}, \boldsymbol{x} + \delta\boldsymbol{x})$ 内的概率密度函数$p(\boldsymbol{x})$ 。

类的原型：概率密度估计

一个模式$\boldsymbol{x}$ 落入区域R的概率

- 对于任意一个模式$\boldsymbol{x}$，其落入区域R的概率P可以表达为：

$$P = \int_{\boldsymbol{x}' \in R} p(\boldsymbol{x}') d\boldsymbol{x}'$$

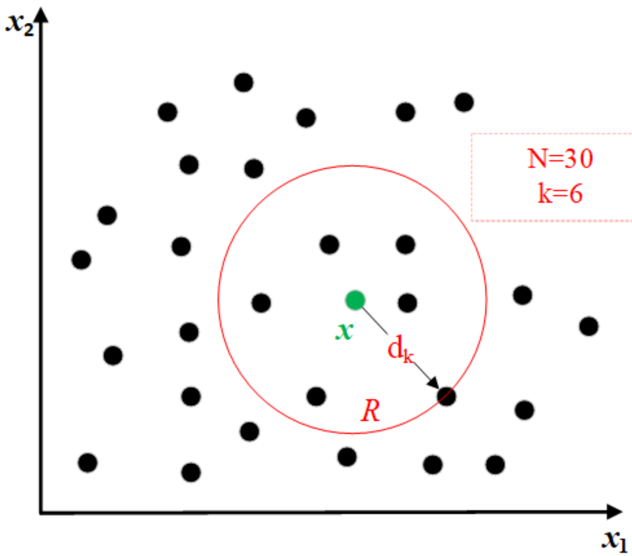
- 如果区域R足够小，P是$p(\boldsymbol{x})$的平滑版本，可以用来估计$p(\boldsymbol{x})$。

类的原型：概率密度估计

k个样本落在区域R

- 但是，由于N非常大，区域R内可能不止落入$\boldsymbol{x}$一个模式！
- 因此，k个样本落在区域R的概率密度可以用二项分布来表达：

$$p(k)=C_N^k P^k(1-P)^{N-k}$$



注意：二项分布p(k)是关于k为自变量的分布，描述的是k取各种值的可能性。

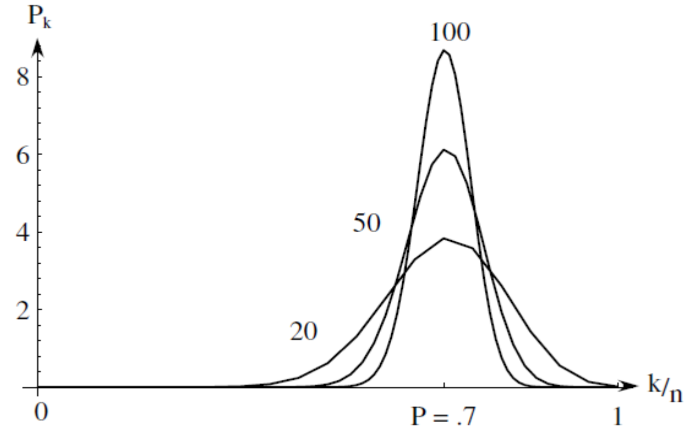
类的原型：概率密度估计

k个样本落在区域R的概率密度

$$p(k) = C_N^k P^k (1-P)^{N-k}$$

- 从二项分布的曲线可以看出，当N很大时， k的分布非常尖锐且集中在均值μ_k附近。
- 二项分布的均值μ_k：

$$E[p(k)] = \mu_k = NP$$



类的原型：概率密度估计

$p(\boldsymbol{x})$的近似估计

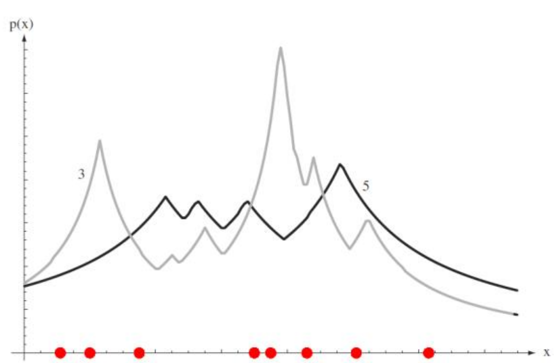
- 因此，当N非常大时，我们可以用二项分布的均值来近似表达k的分布： $k \rightarrow \mu_k$
- 以此得到P的近似估计： $k \approx \mu_k = NP \Rightarrow P = \dfrac{k}{N}$
- 对照刚才对P的定义，并假设在极小的区域R内概率密度$p(\boldsymbol{x})$相同。给定区域R的体积记作V。可以得到$p(\boldsymbol{x})$的近似估计：

$$P = \int_{\boldsymbol{x}' \in R} p(\boldsymbol{x}')d\boldsymbol{x}' \approx p(\boldsymbol{x})V \Rightarrow p(\boldsymbol{x}) \approx \frac{k}{NV}$$

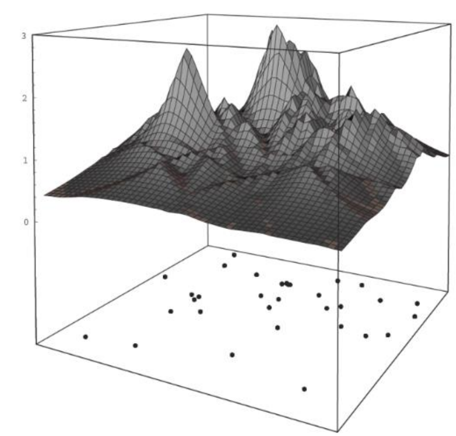
类的原型：概率密度估计

- 如何确定V？ $p(\mathbf{x}) \cong \frac{k/n}{V}$
  - 思路：把V当做训练样本的函数。区域R以$\boldsymbol{x}$为中心，并不断扩张到可以囊括k个训练样本，即k是训练样本个数N的函数。
    - 如果概率密度在$\boldsymbol{x}$附近很高，则会有较多的训练样本落在$\boldsymbol{x}$附近，则k较小；
    - 如果概率密度在$\boldsymbol{x}$附近很低，则会有较少的训练样本落在$\boldsymbol{x}$附近，则k较大。

8个训练样本，k=3、k=5



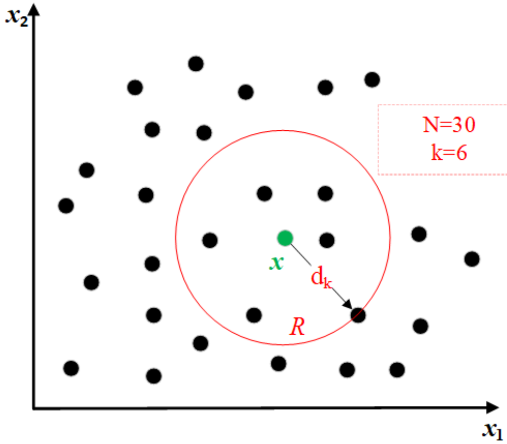
k=5



KNN估计

- K近邻（k-nearest neighbor, KNN）估计：
  - ✓ 给定N个训练样本和k值。
  - ✓ 以任意模式$\boldsymbol{x}$为中心，找到区域R使其包含k个训练样本。
  - ✓ 第k个样本与$\boldsymbol{x}$的距离记作$d_k(\boldsymbol{x})$，则体积$V = 2d_k(\boldsymbol{x})$。
  - ✓ 概率密度估计表达为：

$$p(\boldsymbol{x}) \approx \frac{k}{2Nd_k(\boldsymbol{x})}$$

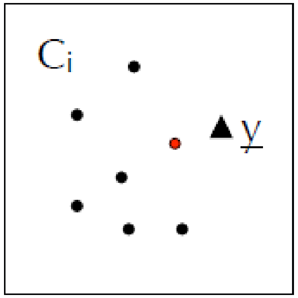


类的原型：最近相邻点

- 最近相邻点(nearest neighbor)：基于KNN估计，设置k=1。
  - 从一类的训练样本中，选取与测试样本距离最近的一个训练样本，作为该类的原型。类的原型取决于测试样本。
    - 表达类的误差较大。
    - 对噪声和异常样本比较敏感。
    - 实际识别过程：
    - （1）选取与测试样本最相邻的训练样本；
    - （2）该训练样本所属的类别，就是测试样本所属的类别。

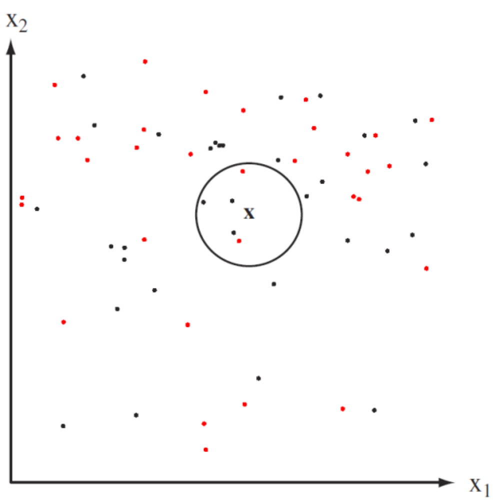
$$\text{Choose } \mathbf{z}_i = \mathbf{x}, \text{ if } d(\mathbf{x}, \mathbf{y}) \text{ is minimized } \forall \mathbf{x} \in C_i$$

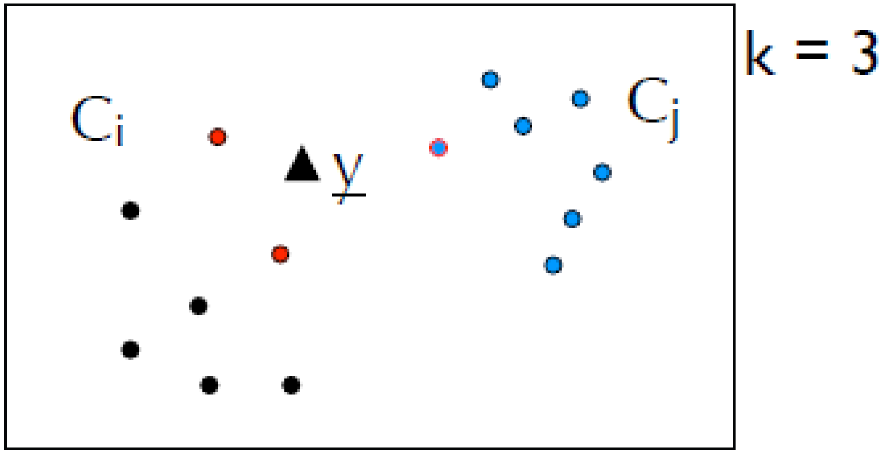
$$\text{if } \mathbf{x} \in C_i, \text{ then classify } \mathbf{y} \text{ as } C_i$$



类的原型：K个最近相邻点

- k个最近相邻点(k-nearest neighbor)：基于KNN估计，设置k>1。
  - 从一类的模式中，选取k个与测试模式距离最近的模式，作为该类的原型。
  - 在这k个样本中，**投票选出**哪个类别的样本最多，测试样本就属于这个类。
  - 特点：部分去除噪声的影响，但计算量较大。





距离测量

- 欧式距离：

$$d_w(\mathbf{x},\mathbf{z}) = \sqrt{\sum_{j=1}^{d}(x_j - z_j)^2} = \sqrt{(\mathbf{x}-\mathbf{z})^T(\mathbf{x}-\mathbf{z})}$$

- Manhattan距离：

$$d(x, z_i) = |x - z_i| = \sum_m |x_m - z_{i,m}|$$

- 距离度量标准(Distance Metric)：
  - ✓ 1. 同一性: $d(x,z)=0, \quad \text{iff} \;\; x = z$
  - ✓ 2. 非负性： $d(x,z)\geq 0$
  - ✓ 3. 对称性： $d(x,z)=d(z,x)$
  - ✓ 4. 三角不等式： $d(x,z)\leq d(x,y)+d(y,z)$

加权欧式距离

- **加权欧式距离：** 对每个维度特征分别设置不同的权重w_j

$$d_w(\mathbf{x},\mathbf{z}) = \sqrt{\sum_{j=1}^{d} w_j (x_j - z_j)^2} = \sqrt{(\mathbf{x}-\mathbf{z})^T \mathbf{I}_w (\mathbf{x}-\mathbf{z})}$$

where $\mathbf{I}_w$ is a diagonal matrix where each diagonal element is w_j.

MED分类器

- MED分类器：
  - 最小欧式距离分类器(Minimum Euclidean Distance Classifier)

- 给定两个类C1和C2，以及两个类各自的原型z1和z2，分类器判别规则如下：

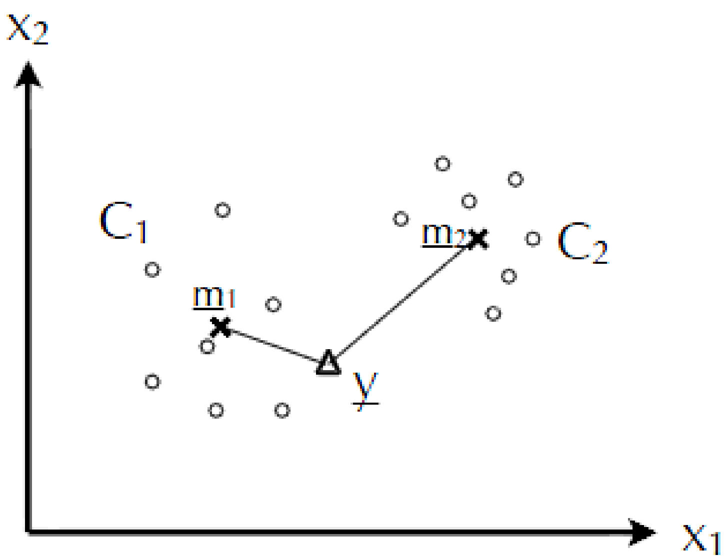
$$\mathbf{x} \in C_1, \text{ if } d(\mathbf{x}, C_1) < d(\mathbf{x}, C_2)$$

$$(\mathbf{x} - \mathbf{z}_1)^T(\mathbf{x} - \mathbf{z}_1) < (\mathbf{x} - \mathbf{z}_2)^T(\mathbf{x} - \mathbf{z}_2)$$

- 类的原型：通常使用均值，也可以使用KNN。
- 距离衡量：欧式距离
- MED分类器也可以用于多类分类。

MED分类器

- MED分类器例子：以均值为类的原型



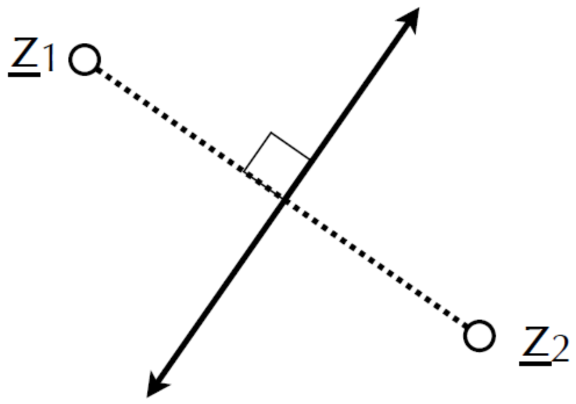
$$d_E(\mathbf{y}, \mathbf{m}_1) < d_E(\mathbf{y}, \mathbf{m}_2)$$

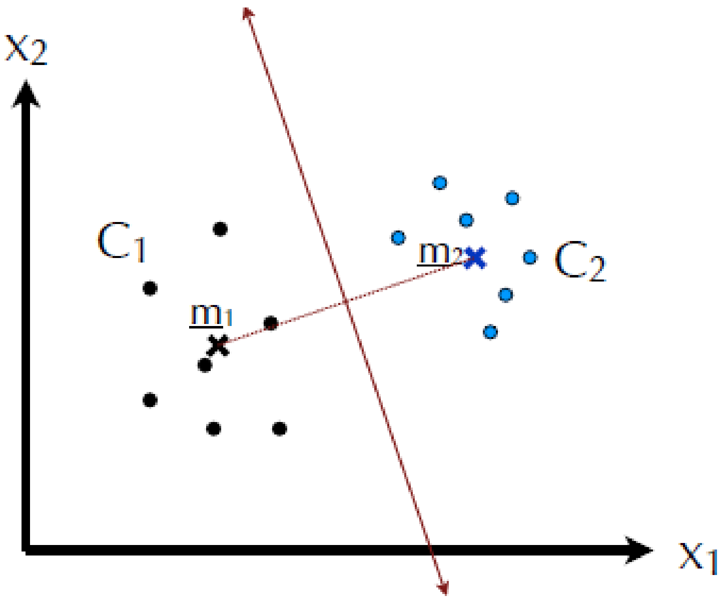
classify pattern $\mathbf{y}$ to class C_1

MED分类器

- 决策边界（分类边界）：通过训练，由分类器判别方程决定的边界。
- 对于2个类而言，MED分类器的决策边界是一个超平面，该平面垂直且二分连接两个类原型的线。

MED决策边界方程：$(\mathbf{x}-\mathbf{z}_1)^T(\mathbf{x}-\mathbf{z}_1)=(\mathbf{x}-\mathbf{z}_2)^T(\mathbf{x}-\mathbf{z}_2)$





MED分类器

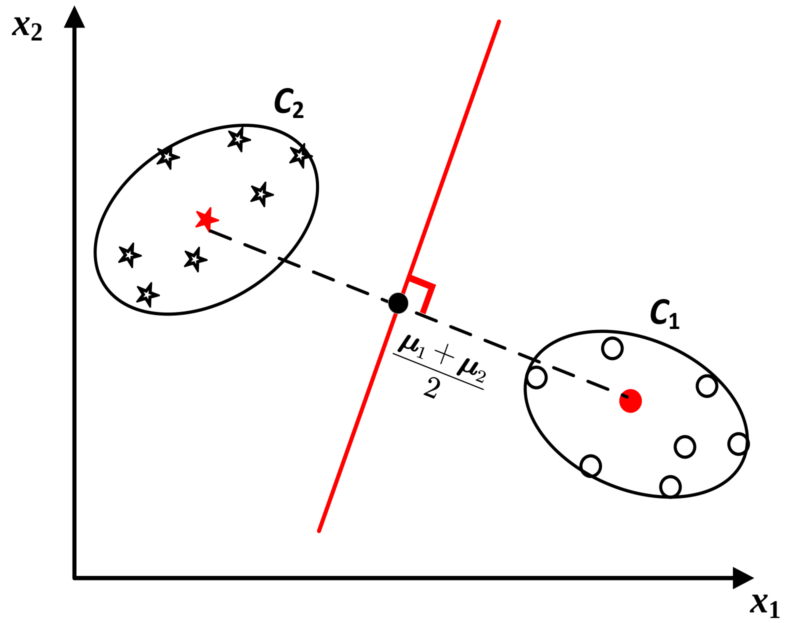
- 对于2个类而言，MED分类器的决策边界方程为：

$$(\boldsymbol{x}-\boldsymbol{\mu}_1)^T(\boldsymbol{x}-\boldsymbol{\mu}_1)-(\boldsymbol{x}-\boldsymbol{\mu}_2)^T(\boldsymbol{x}-\boldsymbol{\mu}_2)=0$$

即
$$(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^T\left(\boldsymbol{x}-\frac{\boldsymbol{\mu}_1+\boldsymbol{\mu}_2}{2}\right)=0$$

关于**x**的一次函数

MED分类器



- 在高维空间中，该决策边界是一个超平面。
- 且该平面垂直且二分连接两个类原型的线。

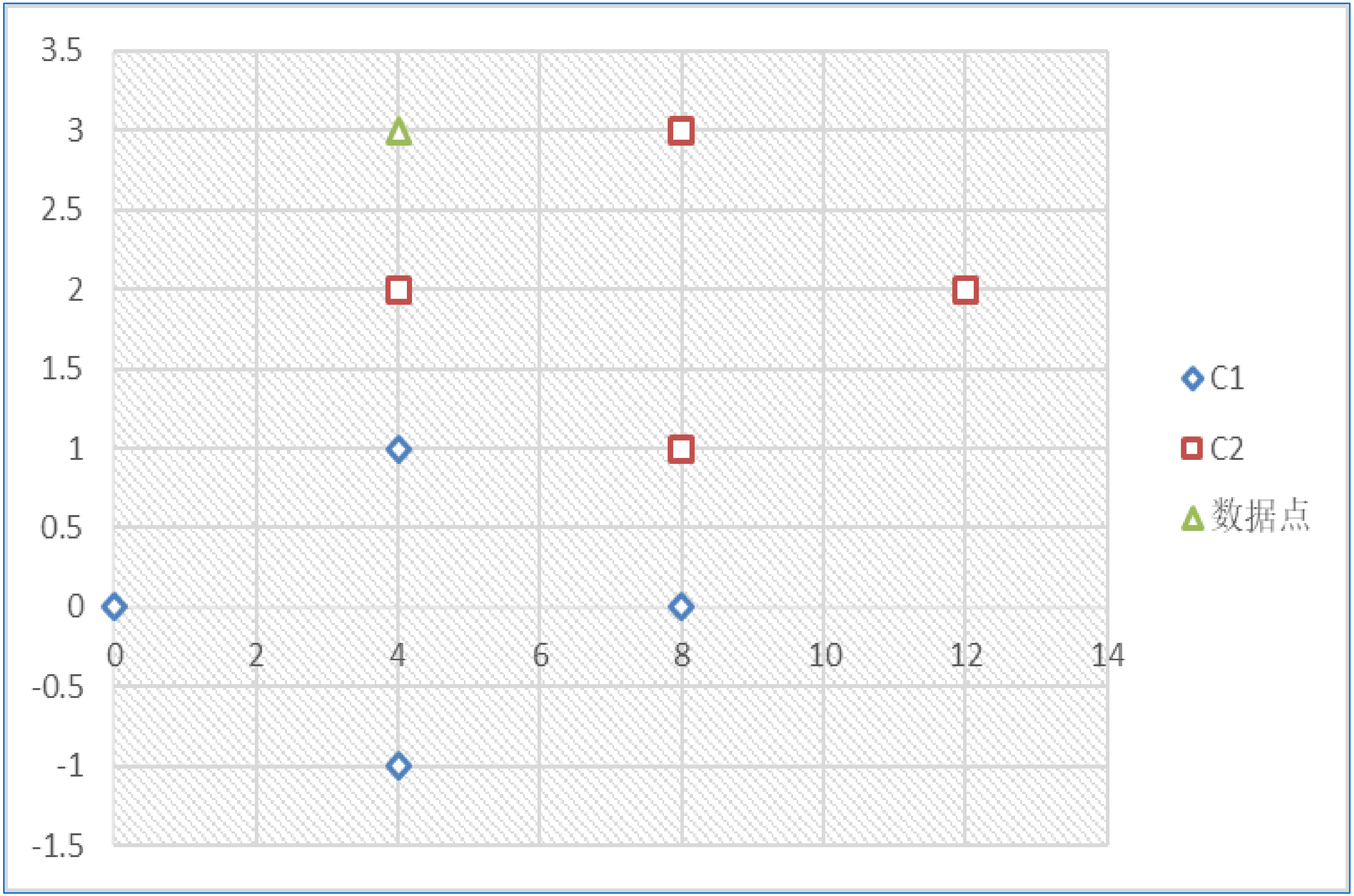
MED分类器

- 例题：

$$C_1 = \{(0,0),(4,1),(4,-1),(8,0)\}$$

$$C_2 = \{(4,2),(8,3),(8,1),(12,2)\}$$

Classify point (4,3) using means as prototypes and MED metric.


3.5
3
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
0
2
4
6
8
10
12
14
C1
C2
数据点

- 上页的例题说明，如果使用均值作为类的原型，MED分类器有时候并不能给出满意的分类性能。
- 如何解决这个问题？尝试一下不同的类原型：
  - **最近相邻点：**(4,2)，所以判断属于C2类
  - **K个最近相邻点：**
    - K=2: (4,2)和(4,1)，分别属于C2和C1类，无法做出判断。
    - K=3: (4,2)、(4,1)、（8,3）或者（4,-1），第三个最近相邻点有两个，而且分属于不同的类，无法做出判断。
    - K=4: (4,2)、(4,1)、（8,3）或者（4,-1），(8,1)：无法判断。
    - K=5: (4,2)、(4,1)、（8,3）或者（4,-1），(8,1)，（8,0）或者(0,0)：无法判断。
    - K=6：(4,2)、(4,1)、（8,3）或者（4,-1），(8,1)，（8,0）或者(0,0)，（12,2）：判断属于C2类。

MED分类器

- 上页的例题，为什么使用基于**均值**的MED分类器不能给出正确的分类呢？
- 看一下两个类各自的特征协方差矩阵：

$$cov(C_1) = \begin{bmatrix} 6.1667 & -3.6667 \\ -3.6667 & 3.3333 \end{bmatrix}$$

$$cov(C_2) = \begin{bmatrix} 10.6667 & 0 \\ 0 & 0.6667 \end{bmatrix}$$

MED分类器

$$cov(C_1) = \begin{bmatrix} 6.1667 & -3.6667 \\ -3.6667 & 3.3333 \end{bmatrix} \qquad cov(C_2) = \begin{bmatrix} 10.6667 & 0 \\ 0 & 0.6667 \end{bmatrix}$$

- ✓ 对角线元素不相等：每维特征的变化不同。
- ✓ 非对角元素不为0：特征之间存在相关性。

- 解决方法：去除特征变化的不同及特征之间的相关性。

特征正交白化

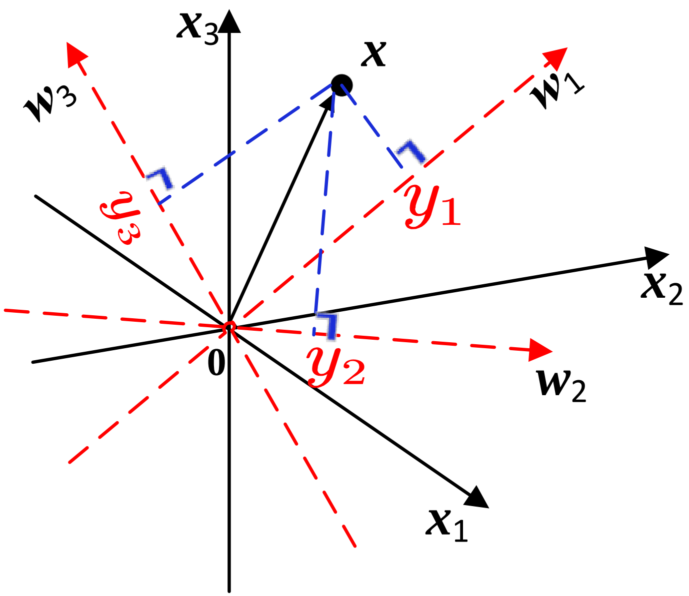
目的

- 将原始特征映射到一个新的特征空间，使得在新空间中特征的协方差矩阵为单位矩阵，从而**去除特征变化的不同**及特征之间的**相关性**。

- 假设映射矩阵为：

$$W = [\boldsymbol{w}_1^T; \boldsymbol{w}_2^T; \ldots; \boldsymbol{w}_p^T]$$

其中$\boldsymbol{w}_i^T$表示新的特征空间的第i个坐标轴。



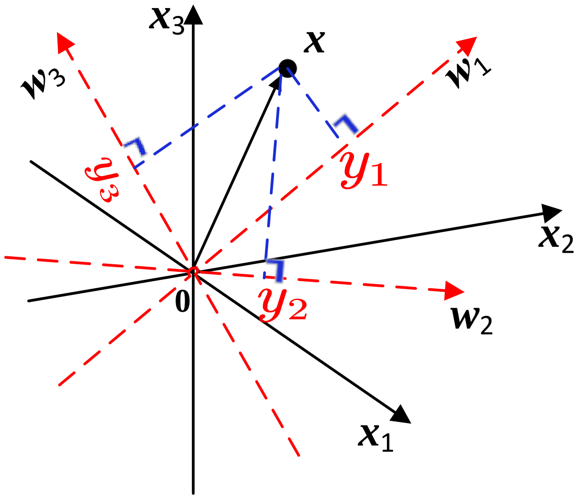
- 转换后的特征$\boldsymbol{y}$为：

$$\boldsymbol{y} = W\boldsymbol{x} = [\boldsymbol{w}_1^T; \boldsymbol{w}_2^T; \ldots; \boldsymbol{w}_p^T]\boldsymbol{x} = \begin{bmatrix} \boldsymbol{w}_1^T\boldsymbol{x} \\ \boldsymbol{w}_2^T\boldsymbol{x} \\ \vdots \\ \boldsymbol{w}_p^T\boldsymbol{x} \end{bmatrix}$$

- 转换后的特征协方差矩阵为：

$$
\begin{aligned}
\Sigma_y = cov(\boldsymbol{y}) &= E[(\boldsymbol{y}-\boldsymbol{\mu}_y)(\boldsymbol{y}-\boldsymbol{\mu}_y)^T] \\
&= E[W(\boldsymbol{x}-\boldsymbol{\mu}_x)(\boldsymbol{x}-\boldsymbol{\mu}_x)^T W^T] \\
&= W\Sigma_x W^T
\end{aligned}
$$

- 目标： $\Sigma_y = I$



- 将特征转换分为两步：先去除特征之间的相关性（解耦, Decoupling），然后再对特征进行尺度变换（白化, Whitening），使每维特征的方差相等。

- 令$W = W_2 W_1$
  - ✓ 解耦：通过W_1实现协方差矩阵对角化，去除特征之间的**相关性**。
  - ✓ 白化：通过W_2对上一步变换后的特征再进行尺度变换，实现所有特征**具有相同方差**。

特征解耦任务

任务

求解转换矩阵W_1，将原始特征构成的协方差矩阵Σ_x对角化。

即 $W_1 \Sigma_x W_1^T = \Lambda$， Λ为对角矩阵。

协方差矩阵Σ_x的对角化

特征向量与特征值

- 如何得到转换矩阵W_1?
  - 第一步：求解协方差矩阵Σ_x的特征值和特征向量。
  - 第二步：由特征向量构建转换矩阵W_1。

- A 是 $p \times p$的矩阵， $\emptyset$ 是 R^p空间的非零向量：
  - 若$A\emptyset = \lambda\emptyset$，则称 $\emptyset$ 是 A的特征向量；
  - λ 是 $\emptyset$ 对应的特征值。

- 特征向量实例：

$$\underline{\phi} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

$$A = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix}$$

$$A\underline{\phi} = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 \\ 6 \end{bmatrix} = 3\underline{\phi}$$

$\underline{\phi}$ 是A的特征向量，其对应的特征值为3。

$$A\boldsymbol{\phi} = \lambda\boldsymbol{\phi} \Rightarrow (A - \lambda I)\boldsymbol{\phi} = 0$$

- 关于特征向量的几点说明：
  - 矩阵A是$(p*p)$方阵。若不是方阵，则需做奇异值分解。
  - 每个特征向量的维数：p维。
  - 如果A是实数对称矩阵，则有p个特征向量和对应的特征值。

$$A\boldsymbol{\phi}_j=\lambda_j\boldsymbol{\phi}_j \quad (j=1,\dots p)$$

特征解耦

- 原始特征协方差矩阵Σ_x的特征向量：$\boldsymbol{v}_1, \boldsymbol{v}_2, \dots, \boldsymbol{v}_p$
- 原始特征协方差矩阵Σ_x的特征值：$\lambda_1, \lambda_2, \dots, \lambda_p$
- 单位化特征向量：$\phi_j = \dfrac{\boldsymbol{v}_j}{||\boldsymbol{v}_j||}$（$j = 1, \dots p$）
- 将所有单位化的特征向量组成矩阵Φ：$\Phi = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \dots, \boldsymbol{\phi}_p]$

$$\Phi^T \Phi = \begin{bmatrix} \boldsymbol{\phi}_1^T \\ \boldsymbol{\phi}_2^T \\ \vdots \\ \boldsymbol{\phi}_p^T \end{bmatrix} [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots, \boldsymbol{\phi}_p] \overset{?}{=} \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix} \quad \Rightarrow \quad \boldsymbol{\Phi}^T = \boldsymbol{\Phi}^{-1}$$

- 由所有特征值组成一个对角矩阵：

$$\Lambda = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_p \end{bmatrix}$$

- 考虑所有特征向量矩阵和所有特征值矩阵相乘：

$$\Phi\Lambda = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots, \boldsymbol{\phi}_p] \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_p \end{bmatrix} = [\lambda_1 \boldsymbol{\phi}_1 \quad \cdots \quad \lambda_p \boldsymbol{\phi}_p]$$

- 考虑所有特征向量矩阵和协方差矩阵相乘：

$$\begin{aligned} \Sigma_x \Phi &= \Sigma_x [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots, \boldsymbol{\phi}_p] = [\Sigma_x \boldsymbol{\phi}_1, \cdots, \Sigma_x \boldsymbol{\phi}_p] \\ &= [\lambda_1 \boldsymbol{\phi}_1 \quad \cdots \quad \lambda_p \boldsymbol{\phi}_p] \end{aligned}$$

- 因此得到：$\Phi\Lambda = \Sigma_x \Phi \Rightarrow \Phi^T \Sigma_x \Phi = \Lambda$

$$\Phi^T = \Phi^{-1}$$

协方差矩阵对角化

- 问题：协方差矩阵的不同特征值对应的特征向量间一定正交吗？
- 对两个不同的特征向量进行如下点积运算：

$$\left.\begin{aligned}\boldsymbol{\phi}_j^T \Sigma_x \boldsymbol{\phi}_i &= \lambda_i \boldsymbol{\phi}_j^T \boldsymbol{\phi}_i \\ \boldsymbol{\phi}_i^T \Sigma_x \boldsymbol{\phi}_j &= \lambda_j \boldsymbol{\phi}_i^T \boldsymbol{\phi}_j\end{aligned}\right\} \xrightarrow{\text{两式相减}} \boldsymbol{\phi}_j^T \Sigma_x \boldsymbol{\phi}_i - \boldsymbol{\phi}_i^T \Sigma_x \boldsymbol{\phi}_j = (\lambda_i - \lambda_j) \boldsymbol{\phi}_j^T \boldsymbol{\phi}_i$$

由于等式中每项都是一个标量：

$$\Rightarrow \boldsymbol{\phi}_j^T \Sigma_x \boldsymbol{\phi}_i = (\boldsymbol{\phi}_j^T \Sigma_x \boldsymbol{\phi}_i)^T = \boldsymbol{\phi}_i^T \Sigma_x^T \boldsymbol{\phi}_j$$

$$\Rightarrow \boldsymbol{\phi}_i^T \Sigma_x^T \boldsymbol{\phi}_j - \boldsymbol{\phi}_i^T \Sigma_x \boldsymbol{\phi}_j = (\lambda_i - \lambda_j) \boldsymbol{\phi}_j^T \boldsymbol{\phi}_i$$

- 协方差矩阵是实对称的，即$\boldsymbol{\Sigma}_{\boldsymbol{x}}^{\boldsymbol{T}} = \boldsymbol{\Sigma}_{\boldsymbol{x}}$：

$$\boldsymbol{\phi}_i^T \Sigma_x^T \boldsymbol{\phi}_j - \boldsymbol{\phi}_i^T \Sigma_x \boldsymbol{\phi}_j = (\lambda_i - \lambda_j)\boldsymbol{\phi}_j^T \boldsymbol{\phi}_i$$

- 上式左边等于0，且$\boldsymbol{\lambda}_{\boldsymbol{i}} \neq \boldsymbol{\lambda}_{\boldsymbol{j}} \Rightarrow \boldsymbol{\phi}_{\boldsymbol{j}}^{\boldsymbol{T}} \boldsymbol{\phi}_{\boldsymbol{i}} = \mathbf{0}$

- 因此不同特征值对应的特征向量正交。

- 协方差矩阵能够通过矩阵 Φ 转换为对角阵，即 Σ_x 矩阵被成功对角化了。

- **W_1：Σ_x的正交单位化的特征向量组成的矩阵。**

$$W_1 = \Phi^T = \begin{bmatrix} \phi_1^T \\ \vdots \\ \phi_p^T \end{bmatrix} \Rightarrow W_1 \Sigma_x W_1^T = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_p \end{bmatrix}$$

转换矩阵W_1的特性

- 将原始特征投影到协方差矩阵对应的特征向量上，每一个特征向量构成一个坐标轴。
- 新特征的协方差是对角阵，对角线上的元素由原协方差矩阵的特征值构成。
- 转换前后欧氏距离保持一致，说明W_1只是起到旋转的作用。

- 证明：原始特征经W_1转换后欧式距离不变。

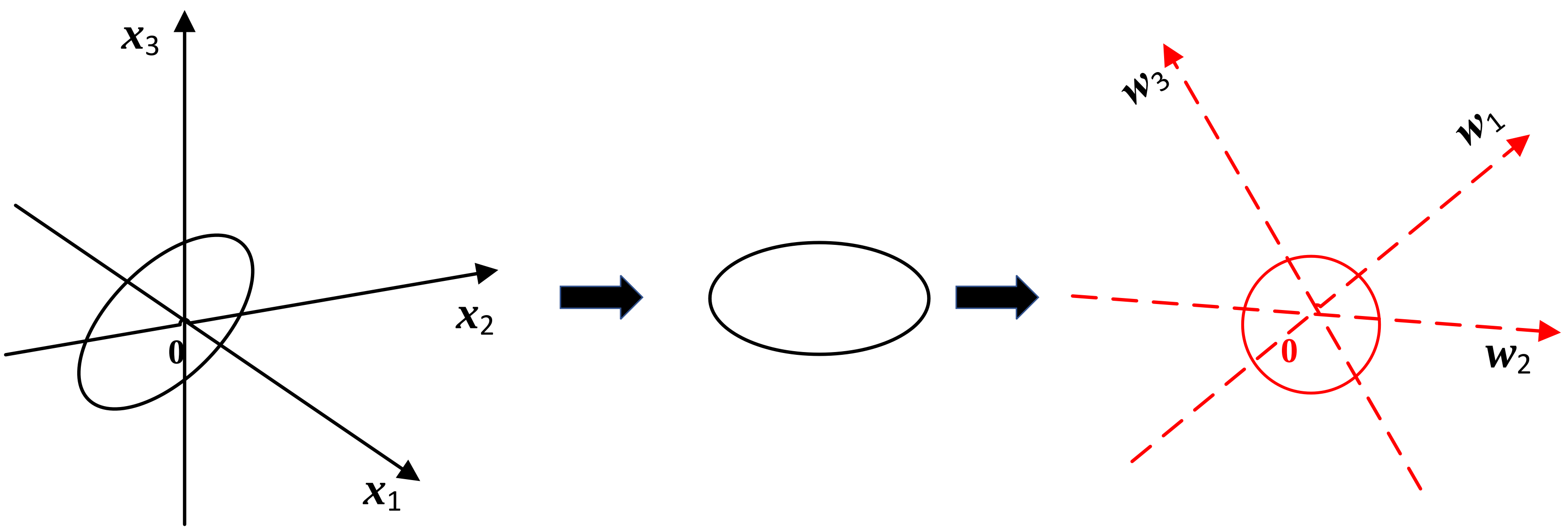
令 $\boldsymbol{z} = W_1\boldsymbol{x} = \Phi^T\boldsymbol{x}$

$$\begin{aligned}
d_E^2(\boldsymbol{z}_1,\boldsymbol{z}_2) &= (\boldsymbol{z}_1-\boldsymbol{z}_2)^T(\boldsymbol{z}_1-\boldsymbol{z}_2)\\
&= \left(\Phi^T(\boldsymbol{x}_1-\boldsymbol{x}_2)\right)^T\Phi^T(\boldsymbol{x}_1-\boldsymbol{x}_2)\\
&= (\boldsymbol{x}_1-\boldsymbol{x}_2)^T\Phi\Phi^T(\boldsymbol{x}_1-\boldsymbol{x}_2)\\
&= (\boldsymbol{x}_1-\boldsymbol{x}_2)^T(\boldsymbol{x}_1-\boldsymbol{x}_2)\\
&= d_E^2(\boldsymbol{x}_1,\boldsymbol{x}_2)
\end{aligned}$$

- W_2的求解：

$$\left.\begin{aligned} W_1\Sigma_x W_1^T &= \Lambda \\ W_2 \Lambda W_2^T &= I \end{aligned}\right\} \Rightarrow W_2 = \Lambda^{-1/2} \qquad \Lambda = \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_p \end{bmatrix}$$

- W的求解：$W = W_2 W_1 = \Lambda^{-1/2}\Phi^T$



- W转换后的欧式距离：

$$
\begin{aligned}
d_E^2(\boldsymbol{y}_1,\boldsymbol{y}_2) &= (\boldsymbol{y}_1-\boldsymbol{y}_2)^T(\boldsymbol{y}_1-\boldsymbol{y}_2) \\
&= \left(\Lambda^{-1/2}\Phi^T(\boldsymbol{x}_1-\boldsymbol{x}_2)\right)^T\Lambda^{-1/2}\Phi^T(\boldsymbol{x}_1-\boldsymbol{x}_2) \\
&= (\boldsymbol{x}_1-\boldsymbol{x}_2)^T\Phi\Lambda^{-1/2}\Lambda^{-1/2}\Phi^T(\boldsymbol{x}_1-\boldsymbol{x}_2) \\
&= (\boldsymbol{x}_1-\boldsymbol{x}_2)^T\Phi\Lambda^{-1}\Phi^T(\boldsymbol{x}_1-\boldsymbol{x}_2) \\
&= (\boldsymbol{x}_1-\boldsymbol{x}_2)^T\left(\Phi\Lambda\Phi^T\right)^{-1}(\boldsymbol{x}_1-\boldsymbol{x}_2) \\
&= (\boldsymbol{x}_1-\boldsymbol{x}_2)^T\Sigma_x^{-1}(\boldsymbol{x}_1-\boldsymbol{x}_2)
\end{aligned}
$$

$$
\boxed{\begin{aligned}
&\Phi^T\Sigma_x\Phi = \Lambda \\
&\Rightarrow \Sigma_x = \Phi\Lambda\Phi^T
\end{aligned}}
$$

- $(\boldsymbol{x}_1-\boldsymbol{x}_2)^T\Sigma_x^{-1}(\boldsymbol{x}_1-\boldsymbol{x}_2)$称为马氏距离(Mahalanobis Distance)。

MICD距离度量

概念

- 最小类内距离分类器（Minimum Intra-class Distance Classifier），基于马氏距离的分类器。
  - ✓ 距离度量：马氏距离；类的原型：均值
- 该距离不仅考虑了类的均值对于距离测量的影响，还加入了**类的方差**对于距离测量的影响。

判别公式

$$\boldsymbol{x} \in C_1, \quad \text{if } d_M(\boldsymbol{x}, C_1) < d_M(\boldsymbol{x}, C_2)$$

$$\text{if } (\boldsymbol{x}-\boldsymbol{\mu}_1)^T \Sigma_1^{-1} (\boldsymbol{x}-\boldsymbol{\mu}_1) < (\boldsymbol{x}-\boldsymbol{\mu}_2)^T \Sigma_2^{-1} (\boldsymbol{x}-\boldsymbol{\mu}_2)$$

马氏距离的属性

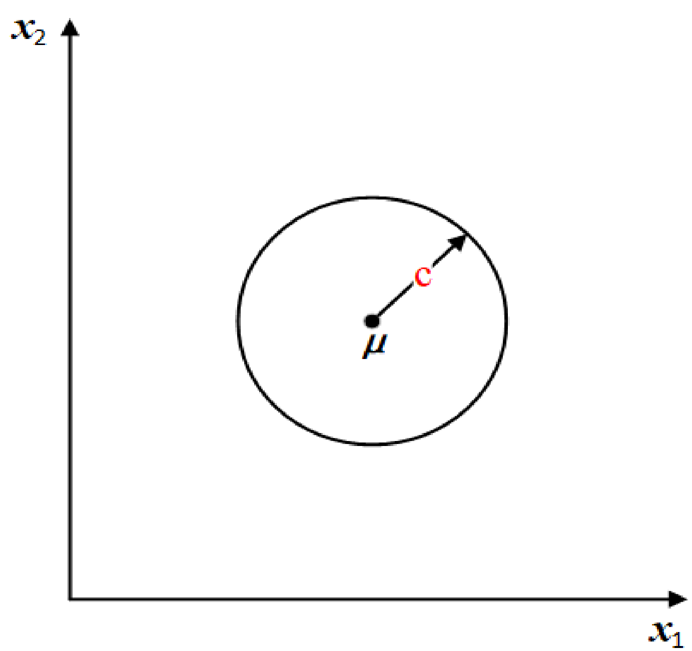
当$\Sigma = I$时：等于欧式距离

$$d_M(\boldsymbol{x}, C_i) = (\boldsymbol{x}-\boldsymbol{\mu}_i)\Sigma_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)$$
$$= (\boldsymbol{x}-\boldsymbol{\mu}_i)^T(\boldsymbol{x}-\boldsymbol{\mu}_i)$$

等距面方程：

$$(\boldsymbol{x}-\boldsymbol{\mu}_i)^T(\boldsymbol{x}-\boldsymbol{\mu}_i) = c^2$$

半径为c的球面。



等距图是一个球面

马氏距离的属性

当Σ为对角矩阵时：

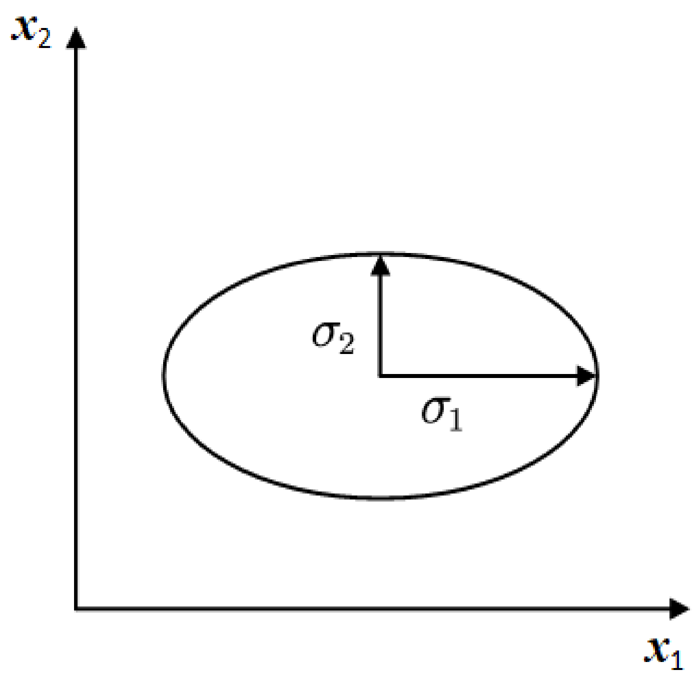
$$d_M(\boldsymbol{x}, C_i) = (\boldsymbol{x} - \boldsymbol{\mu}_i)\Sigma_i^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_i)$$

$$= (\boldsymbol{x} - \boldsymbol{\mu}_i)^T \begin{bmatrix} \frac{1}{\sigma_1^2} & \cdots & 0 \\ \vdots & \cdots & \vdots \\ 0 & \cdots & \frac{1}{\sigma_p^2} \end{bmatrix} (\boldsymbol{x} - \boldsymbol{\mu}_i)$$

$$= \sum_{j=1}^{p} (\frac{x_i - \mu_{ij}}{\sigma_j})^2$$

等距面方程：

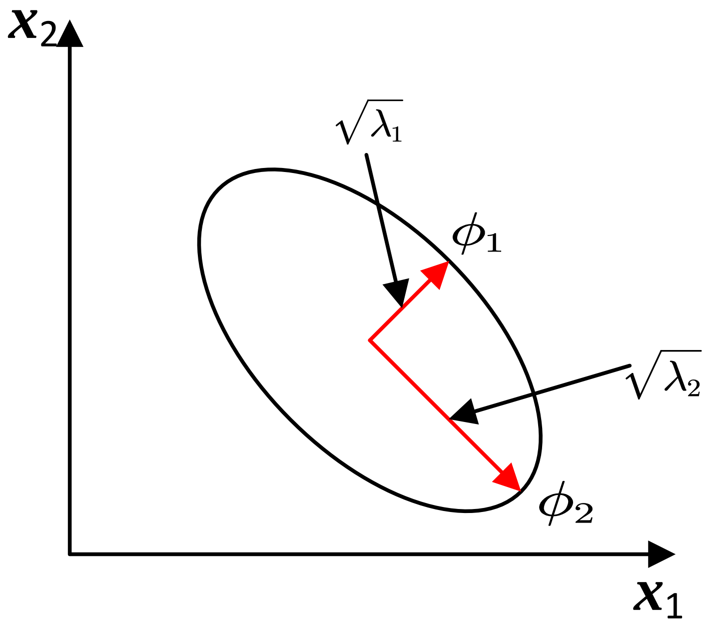
$$\sum_{j=1}^{p} (\frac{x_i - \mu_{ij}}{\sigma_j})^2 \quad = c^2$$



等距图是一个超椭圆面

马氏距离的属性

当Σ是任意值时：等距图是一个有方向的超椭圆面。



ϕ：协方差矩阵 Σ 的特征向量

λ：协方差矩阵 Σ 的特征值

MICD分类器的决策边界

对于二类分类而言，

MICD分类器的决策边界位于到两个类的距离相等的面上，即：

$$(\boldsymbol{x}-\boldsymbol{\mu}_1)^T\Sigma_1^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_1)-(\boldsymbol{x}-\boldsymbol{\mu}_2)^T\Sigma_2^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_2)=0$$

$$\vdots$$

$$\boldsymbol{x}^T(\Sigma_1^{-1}-\Sigma_2^{-1})\boldsymbol{x}+2(\boldsymbol{\mu}_2^T\Sigma_2^{-1}-\boldsymbol{\mu}_1^T\Sigma_1^{-1})\boldsymbol{x}+\boldsymbol{\mu}_1^T\Sigma_1^{-1}\boldsymbol{\mu}_1-\boldsymbol{\mu}_2^T\Sigma_2^{-1}\boldsymbol{\mu}_2=0$$

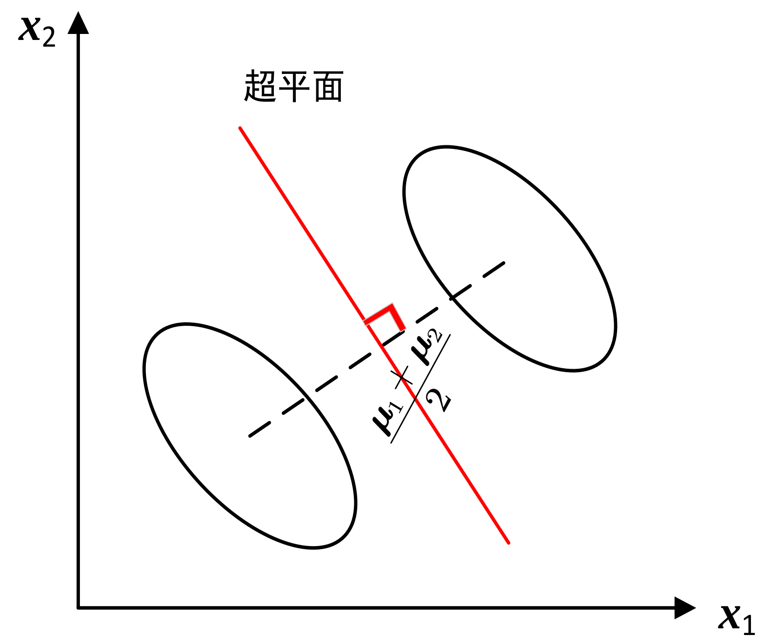
当$\Sigma_1 \neq \Sigma_2$时，关于x的二次函数

MICD分类器的决策边界

当$\Sigma_1 = \Sigma_2$时，决策边界函数可化简为：

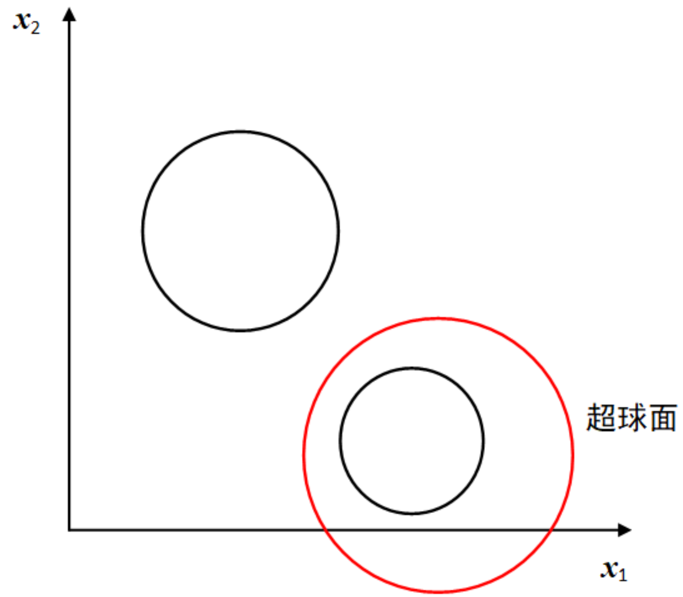
$$(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^T\Sigma^{-1}\left(\boldsymbol{x}-\frac{\boldsymbol{\mu}_1+\boldsymbol{\mu}_2}{2}\right)=0$$

此时决策边界为经过$\frac{\boldsymbol{\mu}_1+\boldsymbol{\mu}_2}{2}$的超平面。



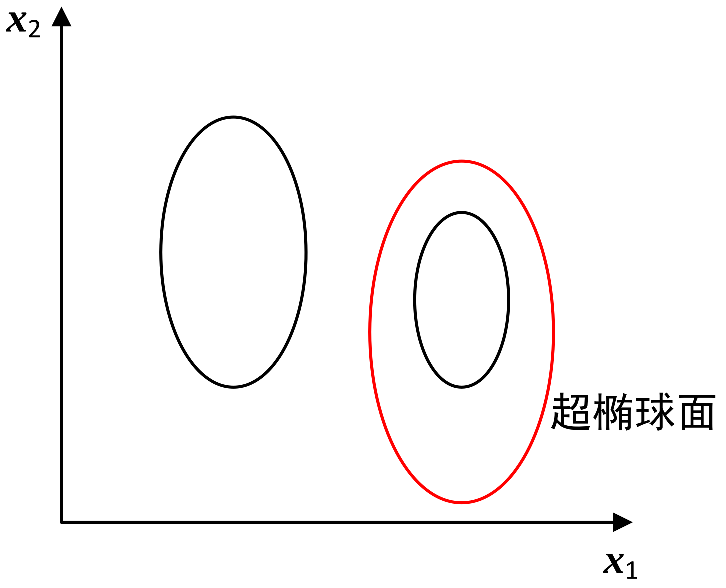
MICD分类器的决策边界

当$\Sigma_1=c_1 I$， $\Sigma_2=c_2 I$ 时，决策边界为超球面。



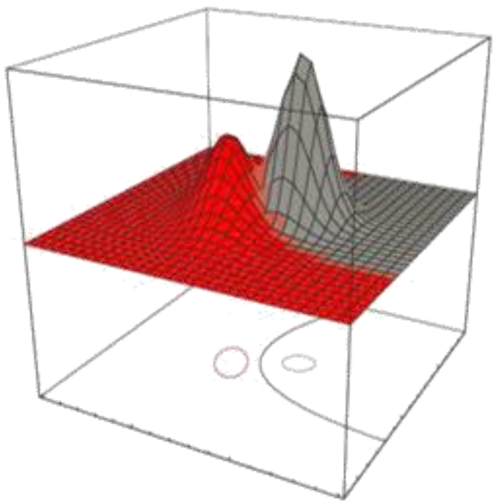
MICD分类器的决策边界

当$\Sigma_1=c\Sigma_2$时，决策边界为超椭球面。

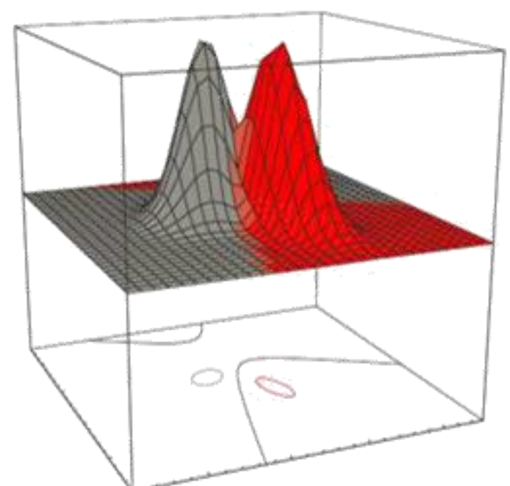


MICD分类器的决策边界

当Σ_1、Σ_2是任意值，决策边界是一个超抛物面或者超双曲面。



超抛物面

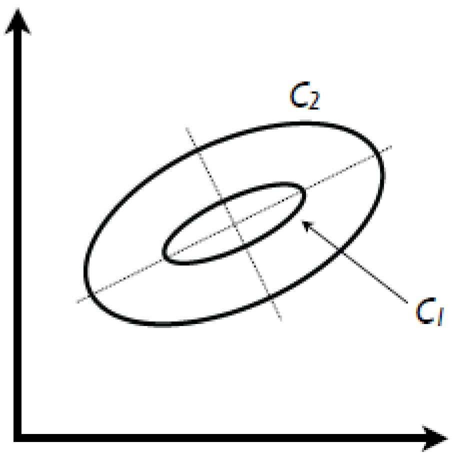


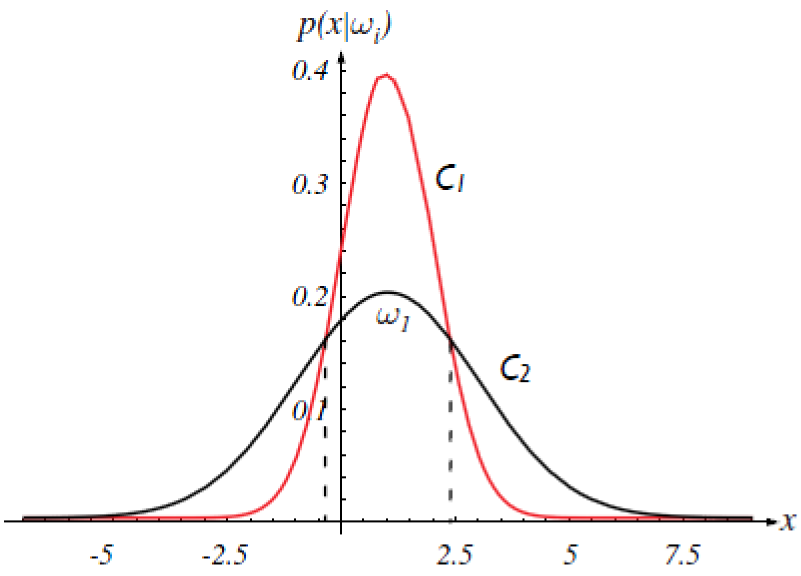
超双曲面

MICD分类器的缺陷

- 如下情况，可以看出MICD分类器的一个缺陷。

$$\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2, \Sigma_1 < \Sigma_2 \ \rightarrow \ d_M(\boldsymbol{y}, C_1) > d_M(\boldsymbol{y}, C_2)$$





$$\text{MICD: } d_M^2(\underline{x}, C_1) > d_M^2(\underline{x}, C_2) \forall \underline{x}$$

MICD分类器会选择方差较大的类