模式识别笔记

1. Introduction to Pattern Recognition

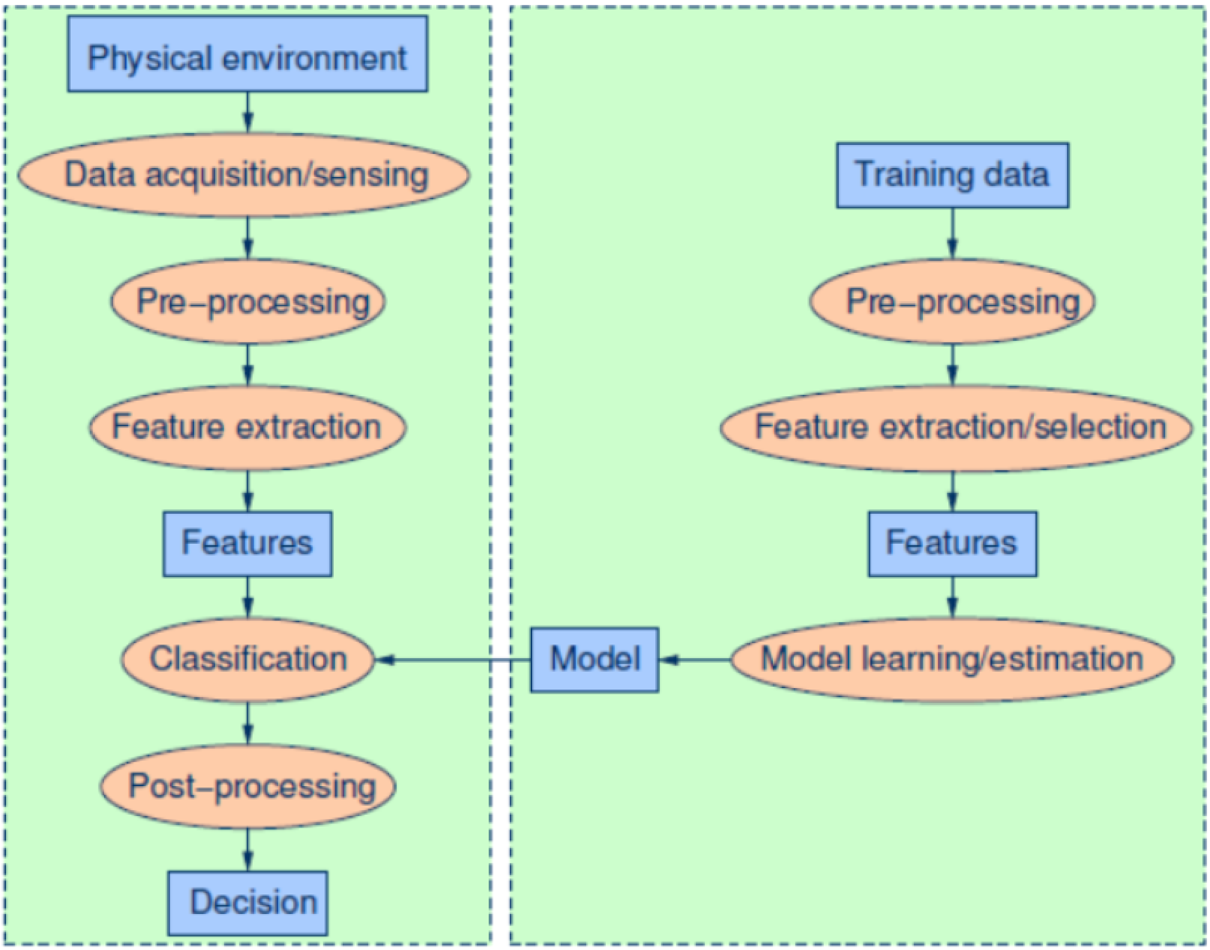
1.1 Machine Perception

> 什么是模式识别?

它是一种研究技术研究机器如何:

- 观察环境
- 学会区分兴趣模式
- 对模式类别作出合理的决定

1.2 模式识别系统



Object/process diagram of a pattern recognition system.

模式识别系统:

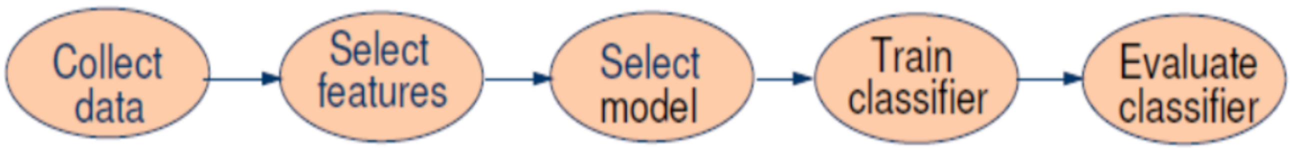
- Data Acquisition & Sensing 【数据获取】

测量物理数据

- pre-procession【 预处理】
  - 去掉噪声数据
  - 从背景中分离感兴趣的模式（分割）
- Feature extraction【 特征抽取】

  在特征方面寻找新的表示
- Model learning / estimation 【模型学习评估】
- Classification 【分类】
- Post-processing 【后期处理】

1.3 Design Cycle



The design cycle.

评估分类器：

- 独立运行 also called Bootstrap

  随机划分训练集测试集，反复训练和测试n遍，取平均准确率高的
- 交叉验证

  将数据集平均划分成k个子集，k-1个用来做训练集，1个用来做测试集，进行k轮，取平均准确率高的

1.4 评估总结

2. Bayesian Decision Theory

2.1 Bayes Rule

贝叶斯公式：

$$p(\omega|x) = \frac{p(x|\omega)p(\omega)}{p(x)} \tag{1}$$

2.2 Bayes Error Rate

对于贝叶斯分类器：

- if $p(\omega_1|x) > p(\omega_2|x),$ decide ω_1
- Otherwise, decide ω_2

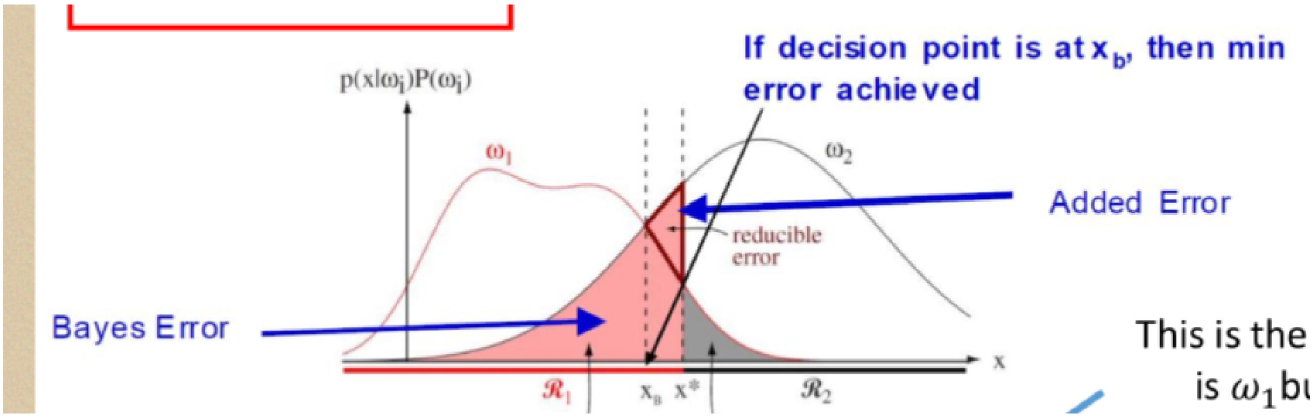
所以2分类分类器误差为：

$$p(error|x) = min[p(\omega_1|x), p(\omega_2|x)] \tag{2}$$

所以n分类为：

$$p(error|x) = 1 - max[p(\omega_1|x), p(\omega_2|x, \dots, p(\omega_n|x))] \tag{3}$$

直观展示：



用积分的思想：

$$\begin{aligned} p(error) &= p(x \in R_2, \omega_1) + p(x \in R_1, \omega_2) \\ &= p(x \in R_2|\omega_1)p(\omega_1) + p(x \in R_1|\omega_2)p(\omega_2 2) \\ &= \int_{R_2} p(x|\omega_1)p(\omega_1)dx + \int_{R_1} p(x|\omega_2)p(\omega_2)dx \end{aligned} \tag{4}$$

决策边界不在较点，则会产生reducible error(可还原误差)

2.3 损失函数

假定：

- c个分类{ω_1,ω_2,...,ω_c}
- a个可能的操作$\{\alpha_1, \alpha_2, \dots, \alpha_a\}$,比如选择去看病or不去(课程中的例子)
- $\lambda_{ij} = \lambda(\alpha_i|\omega_i)$表示分类是$\omega_j$的时候采取操作$\alpha_i$带来的损失

对于给定的观察状态x，若它的真实分类为w_j，而我们选择了操作α_i，其损失则是λ_{ij}

进一步的，对于所有的可能的状态，对于选择了操作α_i其损失(**Conditional risk**)为：

$$R(\alpha_i|x) = \sum_j^c \lambda_{ij} p(\omega_j|x) \tag{5}$$

自然的，对于所有可能的观察，总体误差(**overall risk**)为：

$$err = \int R(\alpha(m)|x)p(x)dx \text{ for } m \in [1, a] \tag{6}$$

显然只要$R(\alpha_i|x)$d达到最小，则总体误差最小。

举个例子，对于一个二分类问题：

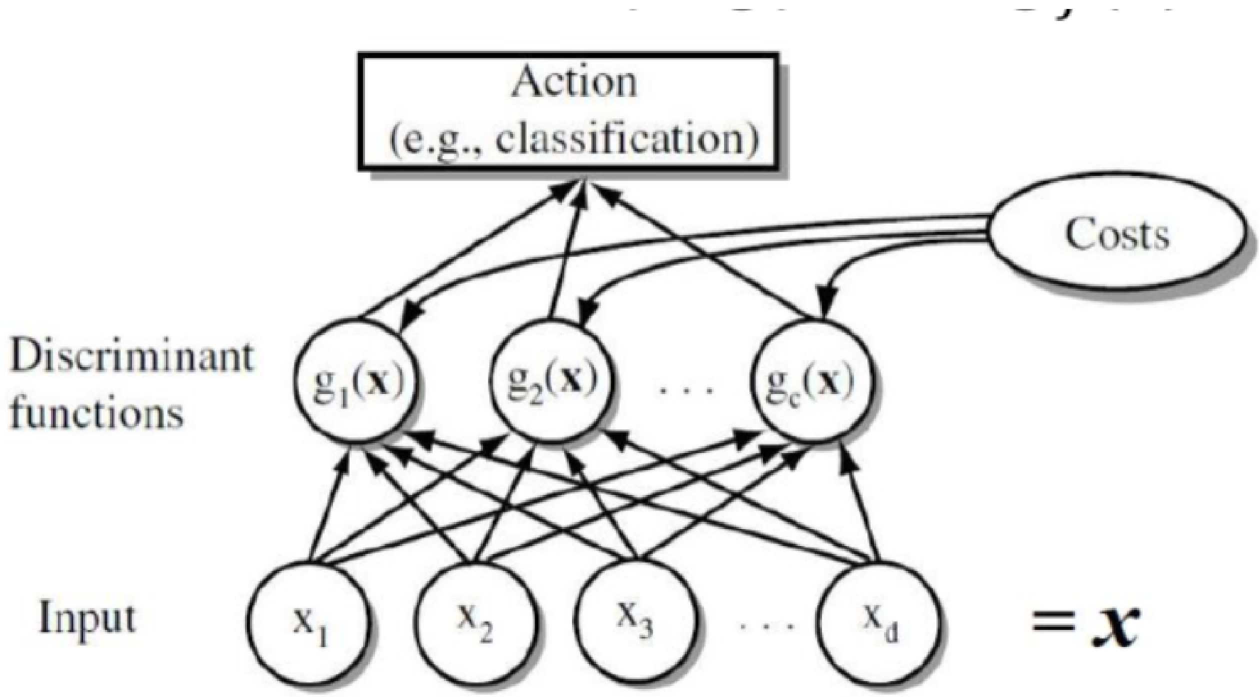
$$R(\alpha_1|x) = \lambda_{11}p(\omega_1|x) + \lambda_{12}p(\omega_2|x)$$
$$R(\alpha_2|x) = \lambda_{21}p(\omega_1|x) + \lambda_{22}p(\omega_2|x)$$

2.4 判别式函数Discriminant Function

对于贝叶斯分类器，可以把它视作是一组判别式函数的集合（共c个判别器，代表一个类一个）：

$$g_i(x), i = 1, \ldots, c$$

如果 $g_i(x) > g_j(x)$ for all $j \neq i$,状态x会被归为类别 ω_i。



当然判定函数的选择不唯一：对于上述的集合，可以定义一个单调递增函数 G：

$$G(g_i(x)) > G(g_j(x)) \text{ if } g_i(x) > g_j(x) \text{ for all } j \neq i$$

例如，可以是log函数:

$$\begin{aligned} G(g_i(x)) &= \ln(g_i(x)) \\ &= \ln(p(\omega_i|x)) \\ &= \ln(\tfrac{p(x|\omega_i)p(\omega_i)}{p(x)}) \\ &= \ln(p(x|\omega_i)) + \ln(p(\omega_i)) - \ln(p(x)) \end{aligned} \tag{7}$$

其中$p(x|\omega_i)$的p可以是高斯分布(即正态分布)

2.5 正态分布 Normal Distribution

先上正态分布公式:

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp[-\frac{1}{2}(\frac{x-\mu}{\sigma})^2] \tag{8}$$

其中σ是标准差，μ是期望。

推广到多维度，对于维度d:

$$
\begin{aligned}
&p(x) = \frac{1}{(2\pi)^{\frac{d}{2}} |\sum|^{\frac{1}{2}}} \exp[-\tfrac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)] \\
&\text{Where} \\
&x = (x_1, x_2, \ldots, x_d)^T \\
&\mu = (\mu_1, \mu_2, \ldots, \mu_d)^T \\
&\Sigma = \textstyle\int (x-\mu)(x-\mu)^T p(x)dx
\end{aligned} \tag{9}
$$

2.5.1 多元正态密度函数下的判别函数

回顾一下我们的判别函数(公式(7)):

$$
g_i(x) = \ln(p(x|\omega_i)) + \ln(p(\omega)) \text{ with } p(\omega_i) \text{ is ignored}
$$

则基于多元正态密度函数下的判别函数为:

$$
\begin{gathered}
g_i(x) = -\frac{1}{2}(x-\mu_i)^T\Sigma^{-1}(x-\mu_i) - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln|\Sigma_i| + \ln(p(\omega_i)) \\
\text{if } p(x|\omega_i) \sim N(\mu_i, \Sigma_i)
\end{gathered} \tag{10}
$$

- 假设对于所有类别的数据，协方差相同，即$\Sigma_i = \Sigma$

  则判别函数(10)可以简化为:

  $$
  g_i(x) = -\frac{1}{2}(x-\mu_i)^T{\Sigma_i}^{-1}(x-\mu_i) + \ln p(\omega_i) \tag{11}
  $$

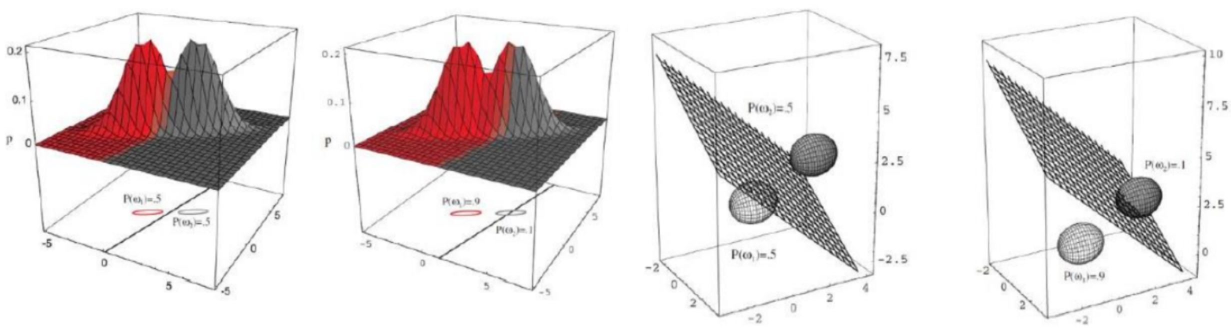
  进一步的，对于公式(11)，前半项拆分:

  $$
  -\frac{1}{2}(x-\mu_i)^T{\Sigma_i}^{-1}(x-\mu_i) = -\frac{1}{2}{\Sigma_i}^{-1}(x^Tx - 2\mu_i x + \mu_i^T\mu_i)
  $$

  注意到 $x^T{\Sigma_i}^{-1}x$ 独立于 i ，可以忽略，因此公式(11)可以进一步化简:

  $$
  \begin{aligned}
  g_i(x) &= {\Sigma_i}^{-1}\mu_i x - -\tfrac{1}{2}{\Sigma_i}^{-1}\mu_i^T\mu_i + \ln(\omega_i) \\
  &= w_i^T x + w_{i0}
  \end{aligned} \tag{12}
  $$

  可以看到这其实是一个线性判别函数，在样本空间里直观地感受下:



- 假设协方差不同

  则:

$$
\begin{aligned}
g_i(x) \quad &= x^T W_i x + w_i x + w_{i0} \\
\text{where} \quad & \\
W_i \quad &= -\frac{1}{2}\Sigma_i^{-1}, \\
w_i \quad &= \Sigma_i^{-1}\mu_i, \\
w_{i0} \quad &= -\frac{1}{2}\mu^T\Sigma_i^{-1}\mu - \frac{1}{2}\ln|\Sigma_i| + \ln p(\omega_i)
\end{aligned} \tag{13}
$$

2.6 极大似然估计 Maximum Likelihood

上一小节假定样本对于每种类概率分布遵循高斯分布，则公式(12)的有两个参数需要估计，分别是$\sum_i$和μ_i。即我们需要估计$p(x|\omega_i)$这一高斯分布(即正态分布)的参数，从而根据一个观察值 x，我们能迅速知道其最可能所属类别。

假定某种分布的优势在于：**把问题从估计某种未知的后验函数简化为估计已知分布函数的参数**

极大似然估计的优势：

- 简单
- 在样本量增加时能够收敛

我们假定:

- 样本集合 $\{x_j\} = D$ 中的每个样本独立同分布，基于概率函数 $p(x|\omega_j)$
- $p(x|\omega_j) \sim N(\mu_j, \Sigma_j)$ ，即服从正态分布

则$p(x|\omega_j) = p(x|\omega_j, \theta_j)$ where $\theta_j = (\mu_j, \Sigma_j)$其中$\theta_j$维度与总类别个数有关，即$(j = 1, 2, \ldots, c)$

我们的目标：**使用 n 个样本来估计参数 θ_j**

基于上面的假设，由于 D 由 n 个独立的样本组成，则有：

$$
p(D|\theta) = \prod_{k=1}^{n} p(x_k|\theta) \tag{14}
$$

其中：

- $p(D|\theta)$ 称作 θ 关于样本的可能性。
- 极大似然估计对于 θ 的估计即是最大化 $p(D|\theta)$
- 根据贝叶斯决策理论，最大化后验概率 $p(x_k|\theta)$ 将产生最小的误差

公式(13)的连乘难以处理，并且有可能浮点溢出，可以做一个对数处理：

$$
\begin{aligned}
l(\theta) \quad &= \ln(p(D|\theta)) \\
&= \sum_{k=1}^{n} \ln(p(x_k|\theta))
\end{aligned} \tag{15}
$$

则极大化似然的 θ 即：

$$
\hat{\theta} = arg\max_{\theta} l(\theta) \tag{16}
$$

最优化的一个必要条件：

$$
\begin{aligned}
&\nabla_\theta l = \sum_{k=1}^{n} \nabla_\theta \ln(p(x_k|\theta)) = 0 \\
&\text{where } \nabla_\theta = \left[\frac{\partial}{\partial\theta_1}, \ldots, \frac{\partial}{\partial\theta_p}\right]^T
\end{aligned} \tag{17}
$$

2.6.1 Case: 未知 μ，Σ 已知

即已知 $p(x_i|\mu) \sim N(\mu, \Sigma)$

回顾2.5节我们的正态分布概率密度函数:

$$p(x) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}}\exp\left[-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right]$$

则它的似然函数Log-likelihood：

$$\sum_{k=1}^{n}\ln(p(x_k|\mu)) = \sum_{k=1}^{n}\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu) - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln(|\Sigma|)\right)$$

根据极大化似然估计，最优的 $\hat{\mu}$ 满足:

$$\nabla_\mu \sum_{k=1}^{n} p(x_k|\mu) = \sum_{k=1}^{n}\Sigma^{-1}(x_k - \hat{\mu}) = 0$$

于是:

$$\sum_{k=1}^{n}(x_k - \hat{\mu}) = 0 \Rightarrow \hat{\mu} = \frac{1}{n}\sum_{k=1}^{n}x_k$$

2.6.2 Case: μ 和 σ 均未知

即 $p(x_i|\mu, \sigma^2) \sim N(\mu, \Sigma)$

类似的：

$$\begin{aligned}
\nabla_\mu \textstyle\sum_{k=1}^{n} p(x_k|\mu,\sigma) &= \textstyle\sum_{k=1}^{n}\Sigma^{-1}(x_k - \hat{\mu}) \\
\nabla_\sigma \textstyle\sum_{k=1}^{n} p(x_k|\mu,\sigma) &= \textstyle\sum_{k=1}^{n}\left(-\frac{1}{\hat{\sigma}} + \frac{(x_k-\hat{\mu})^2}{\hat{\sigma}^2}\right)
\end{aligned}$$

则最优的 $\hat{\mu}$ 和 $\hat{\sigma}$ 为：

$$\begin{aligned}
\hat{\mu} &= \tfrac{1}{n}\textstyle\sum_{k=1}^{n}x_k \\
\hat{\sigma}^2 &= \tfrac{1}{n}\textstyle\sum_{k=1}^{n}(x_k-\hat{\mu})^2
\end{aligned} \tag{18}$$

2.6.3 如何使用ML训练分类器

假定:

- 给定训练集 D
- $D = (x_k, y_k)$，其中 $k = 1, 2, \cdots, n$表示数据维度为 n；$y_k = \omega_1, \omega_2, \cdots, \omega_c$表示共 c 个类

方法:

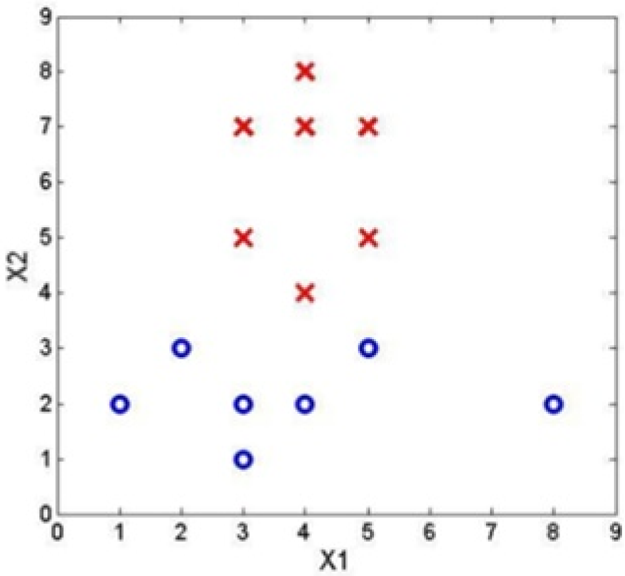
- 将训练集 D 划分为 D_i , 其中 $i = 1, \cdots, c$ ，样本集 D_i 属于类别 ω_i
- 使用每个 D_i 对每个类别分别估计参数 μ_i 和 Σ_i
- $g_i(x)$ 取决于参数 μ_i 和 Σ_i

2.6.4 一个例子

- **Example**

Assume $p(\omega_1) = 0.5, p(\omega_2) = 0.5$

x_1	x_2	y
1	2	1
2	3	1
3	1	1
3	2	1
4	2	1
5	3	1
8	2	1
3	5	2
3	7	2
4	4	2
4	7	2
4	8	2
5	5	2
5	7	2



由公式(17): $\hat{\mu} = \frac{1}{n}\sum_{k=1}^{n} x_k, \hat{\sigma}^2 = \frac{1}{n}\sum_{k=1}^{n}(x_k - \hat{\mu})^2$可知:

$$
\begin{aligned}
\hat{\mu_1} &= (3.71, 2.14)^T, & \hat{\mu_2} &= (4.00, 6.14)^T \\
\hat{\sigma}_1^2 &= (4.49, 0.41)^T, & \hat{\sigma}_2^2 &= (0.57, 1.84)^T \\
\widehat{\Sigma}_1 &= \begin{pmatrix} 4.49 & 0 \\ 0 & 0.41 \end{pmatrix}, & \widehat{\Sigma}_2 &= \begin{pmatrix} 0.57 & 0 \\ 0 & 1.84 \end{pmatrix} \\
\widehat{\Sigma}_1^{-1} &= \begin{pmatrix} 0.22 & 0 \\ 0 & 2.44 \end{pmatrix}, & \widehat{\Sigma}_2^{-1} &= \begin{pmatrix} 1.75 & 0 \\ 0 & 0.54 \end{pmatrix}
\end{aligned}
$$

回顾我们的判别式函数(公式13):

$$
\begin{aligned}
g_i(x) &= x^T W_i x + w_i x + w_{i0} \\
\text{where} & \\
W_i &= -\frac{1}{2}\Sigma_i^{-1}, \\
w_i &= \Sigma_i^{-1}\mu_i, \\
w_{i0} &= -\frac{1}{2}\mu^T\Sigma_i^{-1}\mu - \frac{1}{2}\ln|\Sigma_i| + \ln p(\omega_i)
\end{aligned}
$$

得出

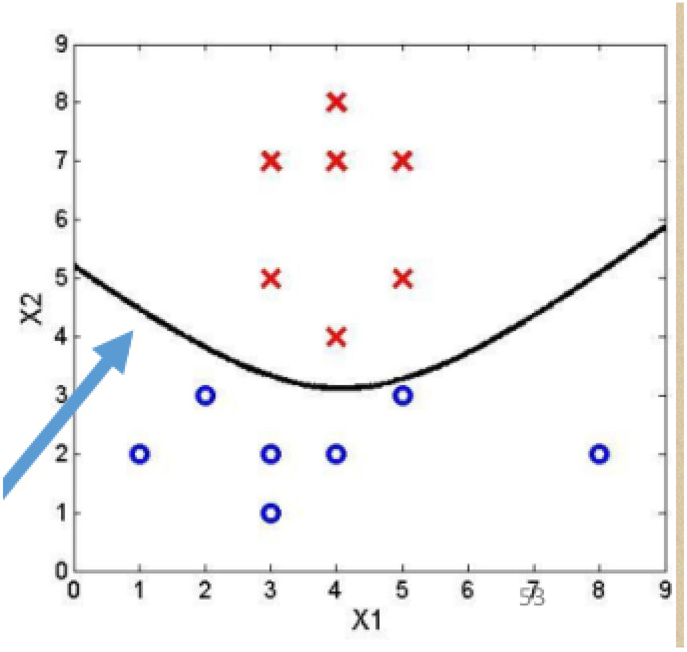
$$
\begin{aligned}
g_1(x) &= -0.11x_1^2 - 1.22x_2^2 + 0.82x_1 + 5.22x_2 - 8.1 \\
g_2(x) &= -0.87x_1^2 - 0.27x_2^2 + 7.02x_1 + 3.34x_2 - 24.9
\end{aligned}
$$

$$
\Downarrow
$$

$$
\text{Decision boundary:}
$$

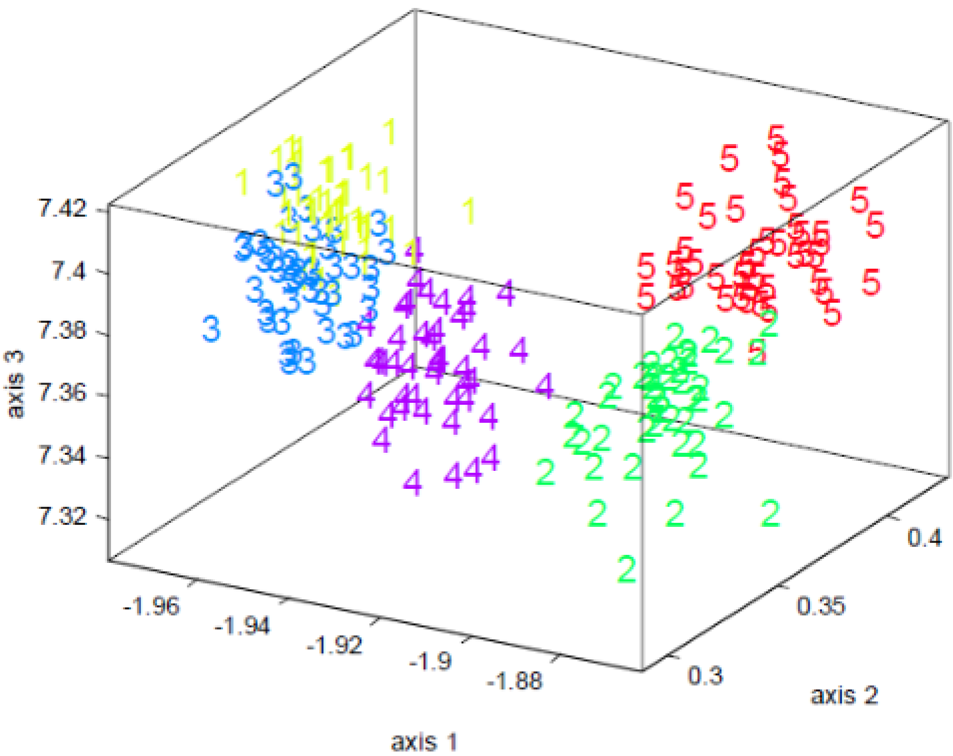
$$
G(x) = g_1(x) - g_2(x) = 0.76x_1^2 - 0.95x_2^2 - 6.20x_1 + 1.88x_2 + 16.8
$$

即



3. 线性模型linear

3.1 参数VS非参数



给定样本集 $(x_i, y_i), i = 1, 2, \cdots, n$,其中 x_i 表示特征向量， y_i 表示样本标签。

考虑一个新的向量 x，要将他分类到可选分类 $C_1, C_2, \cdots, C_c$ 中 。

方法:

- 参数的

- 非参数的

3.1.1 参数方法

参数方法：

- 参数方法假设样本分布的形式(概率密度函数Probability Density Function)是已知的
- 使用训练样本来估计分布参数，比如高斯分布中的 μ 和 σ
- 如果对于分布的假设是正确的，则预测会很准确；否则预测可能会很差

参数方法使用**极大似然估计**来训练分类器，这点在前面一章的贝叶斯决策论也讲过。

假定：

- 给定训练集 $D = (x_k, y_k), k = 1, 2, \cdots, n$
- $p(x|\omega_i) \sim N(\mu_i, \Sigma_i), i = 1, 2, \cdots, c$

方法：

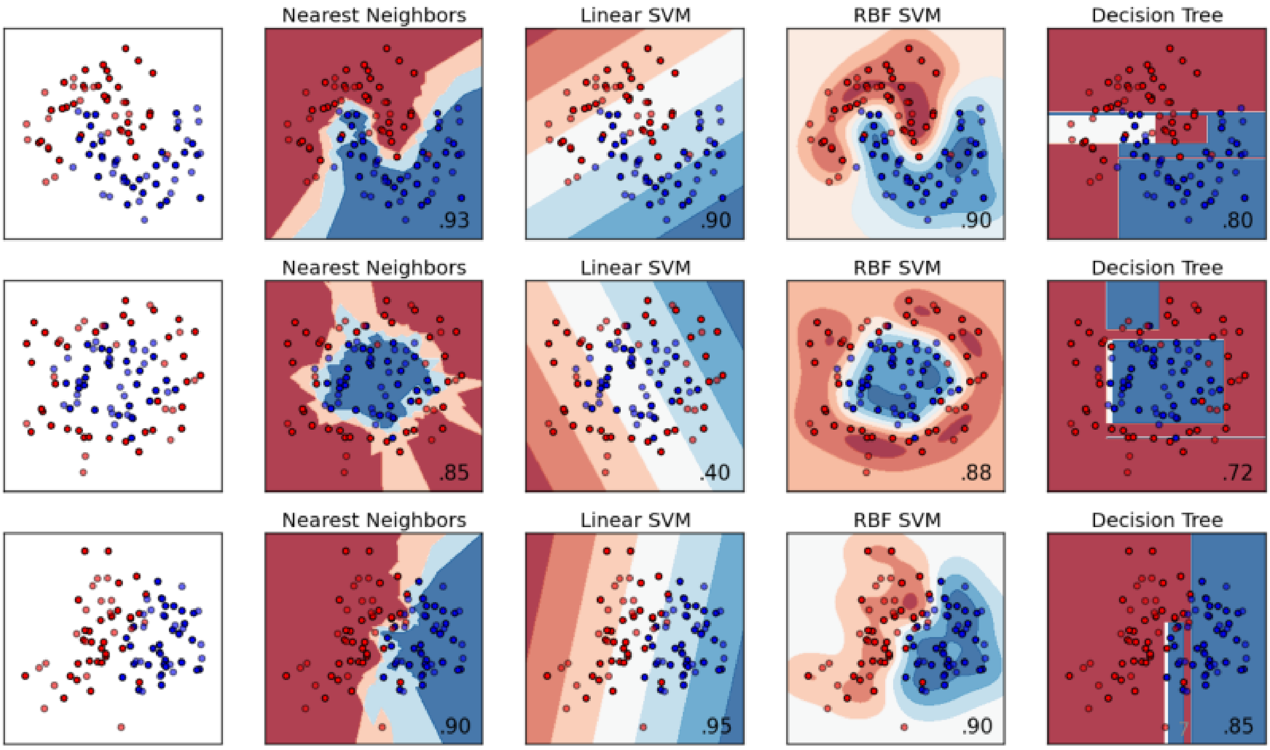
- 将训练集 D 划分为 $D_i, i = 1, 2, \cdots, c$
- 对每一个分类的数据 D_i 分别估计 μ_i 和 Σ_i
- 判别函数 $g_i(x)$ 取决于 μ_i 和 Σ_i

$$\hat{\mu} = \frac{1}{n}\sum_{k=1}^{n} x_k, \qquad \hat{\sigma}^2 = \frac{1}{n}\sum_{k=1}^{n}(x_k - \hat{\mu})^2 \tag{19}$$

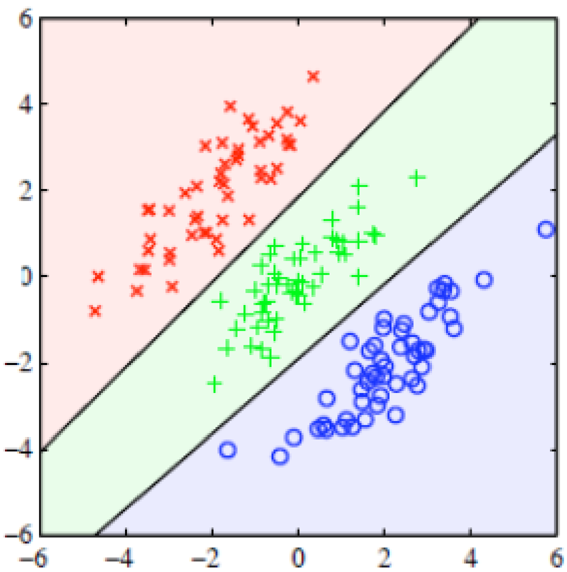
3.1.2 非参数方法

非参数方法：

- 不会去假设样本分布符合某种特定分布
- 相反，它假设判别函数具有某种特定的形式。比如SVM， 神经网络等等
- 训练样本被用来估计分类器的参数
- 局部最优，但易于使用



3.2 线性分类模型(二分类为例)



考虑一个简单的场景：类之间不相交

- 数据线性可分
- 不同类的数据由一个**线性决策表面**完全分开

线性判别式注意：

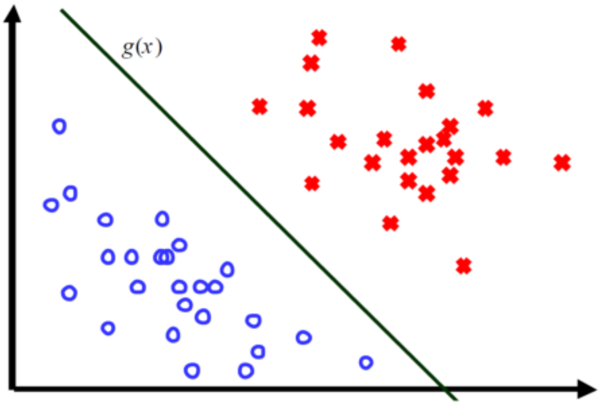
- 决策表面是输入的线性函数
- 输入空间划分为决策区域

3.2.1 二分类问题

决策表面如此定义 $g(x)=0$:

$$g(x)=w^Tx+w_0 \tag{20}$$

因为 $g(x)$ 是线性的，所以决策表面是一个超平面：

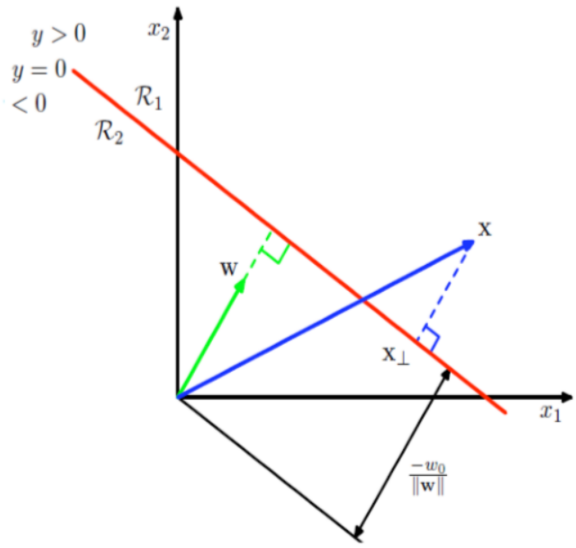


线性判别函数的几何意义

对于一个2分类线性判别函数：

- 判别函数表示向量x(代表一个待分类的数据)各个分量的线性组合，公式(2)已经说明了这个问题
- $g(x)=w^Tx+w_0$，其中 w 和 w_0 表示权重向量和偏置
- 对于一个给定的 x，若 $g(x)\geq 0$,则 $x\in C_1$ ，否则 $x\in C_2$
- 决策边界 $g(x)=0$

几何意义：任意点到决策表面的距离



- 令 x 为任意点
- 令 $x_\perp$ 表示 x 到决策表面的正交投影

$$x=x_\perp+r\frac{w}{||w||} \text{ where } r \text{ denote the distance between } x_\perp \text{ and } x$$

- 两边都乘以相同的因子 w^T,则:

$$g(x) = 0 + \frac{r}{||w||} \Rightarrow r = \frac{g(x)}{||w||}$$

3.2.2 决策区域的凸性

简单的说，就是两个点 x_a 和 x_b在区域 R_k中，则两点连线上的所有点，均在这个区域内。

3.2.3 向量增强

对公式(2)如下操作：

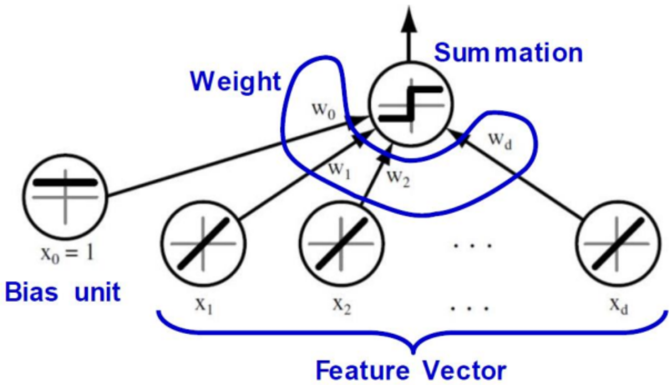
- 增加一维 $x_0 = 1$
- $x \leftarrow (x_0, x)$
- $w \leftarrow (\omega_0, w)$

于是:

$$g(x) = w^T x \tag{21}$$

显然这个决策边界在增强的 $D+1$ 维样本空间中穿过原点

3.2.4 模型小结



- 判别函数表示向量x(代表一个待分类的数据)各个分量的线性组合，视作每个独立单元
- 每个单元都具有输入输出
- 输入单元精确输出与输入相同的值
- 如果加权输入之和大于0，则输出单元输出1，否则输出-1

3.2.5 感知器算法

输入向量 x 通过一个固定的非线性变化得到一个特征向量 $\phi(x)$

$$g(x) = f(w^T \phi(x))$$

f 是一个符号函数

$$f(a) = \begin{cases} +1, a \geq 0 \\ -1, a < 0 \end{cases}$$

+1和-1分别表示向量 x 属于两个类。根据这个设定，我们可以得到损失函数。

感知器标准

使用标签 $y_n \in \{+1,-1\}$，每个模式需要满足:

$$w^T\phi(x_n)y_n > 0$$

对每个分错的样本，感知器标准试图最小化:

$$E(w) = -w^T\phi(x_n)y_n \tag{22}$$

算法流程

随机梯度下降梯度更新公式:

$$\begin{aligned} w^{k+1} &= w^k - \eta\nabla E(w) \\ &= w^k + \eta\phi(x_n)y_n \end{aligned} \tag{23}$$

其中，η 是学习率，k 是steps

算法训练循环以下步骤:

- 如果样本错分为$C_1(y_n=+1)$，增加权重
- 如果样本错分为$C_2(y_n=-1)$，减小权重

3.2.6 最小二乘分类

主要思想，最小化投影距离:

$$J_s(w) = \sum_{i=1}^{n}(w^Tx_i - b_i)^2 \tag{24}$$

其中 b_i 是任意选取的。

对公式(6)进一步化简:

$$J_s(w) = ||Xw-b||^2 \tag{25}$$

其中矩阵符号:

$$w = \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{pmatrix}, X = \begin{pmatrix} x_{10} & x_{11} & \cdots & x_{1d} \\ x_{20} & x_{21} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n0} & x_{n1} & \cdots & x_{nd} \end{pmatrix}, b = \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_d \end{pmatrix}$$

最小化 J, 显然 $Xw=b$，所以有:

$$w = X^{-1}b$$

然而，矩阵 X 可能是奇异的，也就是说，没有逆矩阵。

违逆法pseudo-inverse method

对于公式(7)，求梯度:

$$\nabla J_s(w) = 2X^T(Xw-b)$$

极值必要条件：

$$X^T X w = X^T b$$

可以求得：

$$w = (X^T X)^{-1} X^T b \tag{26}$$

最小均方算法least-Mean-Squared

相比于违逆法，该方法的优势在于

- 违逆法在 $X^T X$ 奇异的时候有问题
- 避免了矩阵很大的时候计算复杂
- 违逆法训练时间更长

回顾公式(6)，直接求其对于w的梯度：

$$\nabla J_s(w) = 2\sum_{i=1}^{n}(w^T x_i - b_i)x_i$$

更新公式：

$$w(k+1) = w(k) + \eta(k)(b_i - w^T x_i)x_i \tag{27}$$

- 即使分离超平面存在，LMS方法也不需要收敛到它
- 由于梯度噪声，LMS不会达到最佳效果

3.2.7 广义线性模型

广义线性判别函数：

$$g(x) = f(w^T x + \omega_0) \tag{28}$$

其中 f 是激活函数。相应的决策表面：

$$g(x) = \text{constant Or } w^T x + \omega_0 = \text{constant}$$

所以决策边界在特征空间里是线性的，即使 f 是非线性的

广义线性判别函数：

$$g(x) = \sum_{i=1}^{n} w_i \phi_i x$$

二次判别函数：

$$\begin{aligned}&(x) = w_0 + w_i x + w_2 x^2\\&\text{where } \phi_1(x) = 1, \phi_2(x) = x, \phi_3(x) = x^2\end{aligned}$$

对于样本 x_i，如果分类正确，则 $g(w, x_i)y_i > 0$

定义一个判别函数 $J(w)$,如果 w是一个解向量，则该函数达到最小值

算法流程：

- 随机选择初始权重 w_1
- 计算梯度 $\nabla J(w(1))$
- 根据负梯度计算：

$$w(k+1) = w(k) - \eta(k)\nabla J(w(k)) \text{ for } \nabla J = \frac{\partial J}{\partial w}$$

  η 是学习率，控制步幅

3.3 多分类

3.3.1 扩展到多分类方法

- One-versus-the-rest

  构建判别函数，使用 c 个分类器，每个分类器解决一个2分类问题
- One-versus-one

  c 个分类，对每两个分类构建一个分类器，则共有 $\frac{c(c-1)}{2}$ 个判别函数

3.3.2 多分类判别

考虑一个 c 分类问题，判别函数形式：

$$y_k(x) = w_k^T x + w_{k0}$$

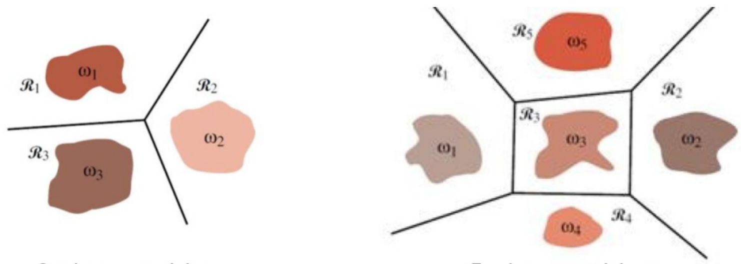
对于给定输入 x，如果 $y_k(x) > y_j(x)$ for all $j \neq k$，则 $x \in C_k$，则 C_k 和 C_j之间的决策边界为：

$$y_k(x) = y_j(x)$$

相应的超平面为：

$$(w_k - w_j)^T x + (\omega_{k0} - \omega_{j0}) = 0$$

二分类问题其实也是如此。



3-class problem　5-class problem

优势：

- 避免默认两可的区域
- 每个决策区域单连通
- 低复杂度

- 需要c个分类器