第 5 章
线性学习机器与线性分类器

5.1 引言

从前面几章我们看到，如果能很好地描述和估计出样本的概率模型，可以用贝叶斯决策或最大似然估计的策略来实现分类或对其他目标进行判定。但是，很多情况下，建立样本的概率模型和准确估计概率密度函数及其他模型参数并不是一件容易的事，在特征空间维数高、内在关系复杂和样本较少的情况下尤其如此。

实际上，模式识别的目的是在特征空间中设法找到两类（或多类）之间的分界面，估计概率密度函数并不是我们的目的。两步贝叶斯决策是首先根据样本进行概率密度函数估计，再根据估计的概率密度函数求分类面。如果能直接根据样本求分类面，就可以省略对概率密度函数的估计。在第 2 章介绍正态分布下的贝叶斯决策时，已经看到，在样本为正态分布且各类协方差矩阵相等的条件下，贝叶斯决策的最优分类面是线性的，两类情况下判别函数形式是 $g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+\omega_0$，而一般情况下为二次判别函数。实际上，如果知道判别函数的形式，可以设法从数据直接估计这种判别函数中的参数。这就是基于样本直接进行分类器设计的思想。进一步，即使不知道最优的判别函数是什么形式，仍然可以根据需要或对问题的理解设定判别函数类型，从数据直接求解判别函数。

基于样本直接设计分类器需要确定三个基本要素，一是分类器即判别函数的类型，也就是从什么样的判别函数类（函数集）中去求解；二是分类器设计的目标或准则，在确定了设计准则后，分类器设计就是根据样本从事先决定的函数集中选择在该准则下最优的函数，通常就是确定函数类中的某些待定参数；第三个要素就是在前两个要素明确之后，如何设计算法利用样本数据搜索到最优的函数参数（即选择函数集中的函数）。形式化表示就是：在判别函数集 $\{g(\alpha),\alpha\in\Lambda\}$ 中确定待定参数 α^*，使得准则函数 $L(\alpha)$ 最小（或最大），即 $L(\alpha^*)=\min\limits_{\alpha} L(\alpha)$。

不同的判别函数类、不同的准则及不同的优化算法就决定了不同的分类器设计方法。

在本章中，我们讨论线性判别函数，即 $g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+w_0$，多类情况下为 $g_i(\boldsymbol{x})=\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{x}+w_{i0}$，$i=1,2,\cdots,c$。采用不同的准则及不同的寻优算法就得到不同的线性判别方法。

线性分类器虽然是最简单的分类器，但是在样本为某些分布情况时，线性判别函数可以成为最小错误率或最小风险意义下的最优分类器。而在一般情况下，线性分类器只能是次优分类器，但是因为它简单而且在很多情况下效果接近最优，所以应用比较广泛，在样本有限的情况下有时甚至能取得比复杂的分类器更好的效果。

用线性函数去根据观测拟合变量间存在的依赖关系，是一个已经有两百年多年历史的研究课题，这就是著名的线性回归问题。在讨论线性分类器之前，我们先对线性回归进行简要的回顾。

5.2 线性回归

线性回归是通过数据发现或估计两个或多个变量之间可能存在的线性依赖关系的基本的统计学方法。最早是在 1805 年和 1809 年由法国数学家 Adrien-Marie Legendre(1762—1833)和德国数学家 Carl Friedrich Gauss(1777—1855)提出的[①]。一般认为，比利时数学家、天文学家、社会学家、统计学家、诗人和剧作家 Adolphe Quetelet(1796—1874)关于“平均人”的一系列定量研究为推动线性回归被广泛应用做出了奠基性的历史贡献。线性回归应该是人们最早对用数学方法关于从数据中“学习”规律的研究，可以看作是机器学习最原始的萌芽。

简单线性回归是学习两个取值为标量的随机变量之间的线性关系，如图 5-1 所示。其中一个变量称作响应变量或依赖变量，通常记作 y，另一个变量称作解释变量或独立变量，通常记作 x。线性回归就是通过(x,y)的一系列观测样本，估计它们之间的线性关系

$$y=w_0+w_1x \tag{5-1}$$

也就是估计其中的系数 w_1 和 w_0。当响应变量依赖于多个解释变量时，这种关系就是多元线性回归

$$y=w_0+w_1x_1+\cdots+w_dx_d=\sum_{i=0}^{d}w_ix_i=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} \tag{5-2}$$

统计学把线性回归作为一门专门的学问进行了深入系统的研究，有大量关于回归的方法、理论性质及各种扩展的结论。我们在本书中只回顾其中最基本的估计方法。

在机器学习领域中，线性回归问题可以描述成如下的问题：

假设有训练样本集

$$\{(x_1,y_1),\cdots,(x_N,y_N)\},\quad x_j\in R^{d+1},\quad y_j\in R \tag{5-3}$$

我们设计学习机器的模型为

$$f(\boldsymbol{x})=w_0+w_1x_1+\cdots+w_dx_d=\sum_{i=0}^{d}w_ix_i=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} \tag{5-4}$$

---

① A. M. Legendre. Nouvelles méthodes pour la détermination des orbites des comètes, Firmin Didot, Paris, 1805. “Sur la Méthode des moindres quarrés” appears as an appendix.

C. F. Gauss. Theoria Motus Corporum Coelestium in Sectionibus Conicis Solem Ambientum. (1809).

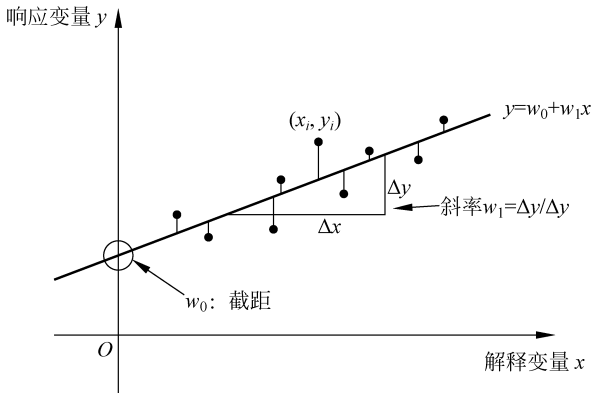


图 5-1 线性回归示意图

其中 $\boldsymbol{w}=[w_0, w_1, \cdots, w_d]^{\mathrm{T}}$ 是模型中待定的参数。线性回归估计的问题就是用训练样本集估计模型中的参数，使模型在最小平方误差意义下能够最好地拟合训练样本，即

$$\min_{\boldsymbol{w}} E = \frac{1}{N}\sum_{j=1}^{N}(f(\boldsymbol{x}_j)-y_j)^2 \tag{5-5}$$

这个目标函数可以写成如下的矩阵形式

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{j=1}^{N}(f(\boldsymbol{x}_j)-y_j)^2 = \frac{1}{N}\|X\boldsymbol{w}-\boldsymbol{y}\|^2 = \frac{1}{N}(X\boldsymbol{w}-\boldsymbol{y})^{\mathrm{T}}(X\boldsymbol{w}-\boldsymbol{y}) \tag{5-6}$$

其中，$X=\begin{bmatrix}\boldsymbol{x}_1^{\mathrm{T}}\\ \vdots \\ \boldsymbol{x}_N^{\mathrm{T}}\end{bmatrix}$ 为全部训练样本的解释变量向量组成的矩阵，$\boldsymbol{y}=\begin{bmatrix}y_1\\ \vdots \\ y_N\end{bmatrix}$ 是全部训练样本的响应变量组成的向量。

使式(5-6)的目标函数最小化的参数 w 应该满足

$$\frac{\partial E(\boldsymbol{w})}{\partial \boldsymbol{w}} = \frac{2}{N}X^{\mathrm{T}}(X\boldsymbol{w}-\boldsymbol{y}) = 0 \tag{5-7}$$

即

$$X^{\mathrm{T}}X\boldsymbol{w} = X^{\mathrm{T}}\boldsymbol{y} \tag{5-8}$$

因此，当矩阵$(X^{\mathrm{T}}X)$可逆时，最优参数的解为

$$\boldsymbol{w}^* = (X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}\boldsymbol{y} \tag{5-9}$$

这就是经典的“最小二乘法”线性回归，其中的矩阵$(X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}$也被称作 X 的伪逆(Pseudo-inverse)矩阵，记作 X^{+}。

采用这样的算法，假如真实的样本是服从式(5-2)的物理规律产生的，只是在观测中带有噪声或误差，则最小二乘线性回归就通过数据学习到了系统本来的模型。而在我们对数据背后的物理模型并不了解的情况下，线性回归给出了在最小平方误差意义下对解释变量与响应变量之间线性关系的最好的估计。

在 5.6 节中我们会看到，线性回归也可以通过迭代的方法求解，并且可以用来解决分类问题。

5.3 线性判别函数的基本概念

在第 2 章中，我们已经遇到过在两类情况下判别函数为线性的情况，这里给出它的一般表达式

$$g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+w_0 \tag{5-10}$$

其中，$\boldsymbol{x}$ 是 d 维特征向量，又称样本向量，$\boldsymbol{w}$ 称为权向量，分别表示为

$$\boldsymbol{x}=\begin{bmatrix}x_1\\x_2\\\vdots\\x_d\end{bmatrix},\quad \boldsymbol{w}=\begin{bmatrix}w_1\\w_2\\\vdots\\w_d\end{bmatrix}$$

w_0 是个常数，称为阈值权。对于两类问题的线性分类器可以采用下述决策规则：令

$$g(\boldsymbol{x})=g_1(\boldsymbol{x})-g_2(\boldsymbol{x})$$

则

$$\begin{cases}\text{如果 } g(\boldsymbol{x})>0,\text{则决策 } \boldsymbol{x}\in\omega_1\\ \text{如果 } g(\boldsymbol{x})<0,\text{则决策 } \boldsymbol{x}\in\omega_2\\ \text{如果 } g(\boldsymbol{x})=0,\text{可将 } \boldsymbol{x} \text{ 任意分到某一类，或拒绝}\end{cases} \tag{5-11}$$

方程 $g(\boldsymbol{x})=0$ 定义了一个决策面，它把归类于 ω_1 类的点与归类于 ω_2 类的点分割开来。当 $g(\boldsymbol{x})$为线性函数时，这个决策面便是超平面。

假设 $\boldsymbol{x}_1$ 和 $\boldsymbol{x}_2$ 都在决策面 H 上，则有

$$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_1+w_0=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_2+w_0 \tag{5-12}$$

或

$$\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{x}_1-\boldsymbol{x}_2)=0 \tag{5-13}$$

这表明，$\boldsymbol{w}$ 和超平面 H 上任一向量正交，即 $\boldsymbol{w}$ 是 H 的法向量。一般来说，一个超平面 H 把特征空间分成两个半空间，即对 ω_1 类的决策域 $\mathcal{R}_1$ 和对 ω_2 类的决策域 $\mathcal{R}_2$。因为当 $\boldsymbol{x}$ 在 $\mathcal{R}_1$ 中时，$g(\boldsymbol{x})>0$，所以决策面的法向量是指向 $\mathcal{R}_1$ 的。因此，有时称 $\mathcal{R}_1$ 中的所有 $\boldsymbol{x}$ 在 H 的正侧，相应地，称 $\mathcal{R}_2$ 中的所有 $\boldsymbol{x}$ 在 H 的负侧。

判别函数 $g(\boldsymbol{x})$可以看成是特征空间中某点 $\boldsymbol{x}$ 到超平面的距离的一种代数度量，见图 5-2。

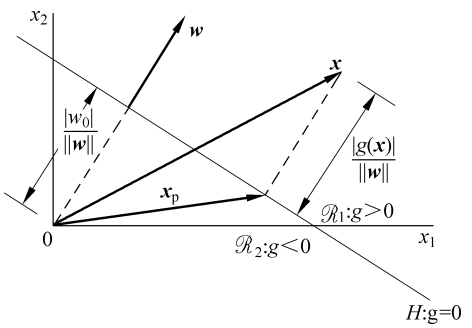


图 5-2 线性判别函数

若把 $\boldsymbol{x}$ 表示成

$$\boldsymbol{x}=\boldsymbol{x}_p+r\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|} \tag{5-14}$$

其中，$\boldsymbol{x}_{\mathrm{p}}$ 是 $\boldsymbol{x}$ 在 H 上的射影向量；r 是 $\boldsymbol{x}$ 到 H 的垂直距离；$\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$是 $\boldsymbol{w}$ 方向上的单位向量。

将式(5-14)代入式(5-10),可得

$$g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\left(\boldsymbol{x}_p+r\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}\right)+w_0=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_p+w_0+r\frac{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}}{\|\boldsymbol{w}\|}=r\|\boldsymbol{w}\|$$

或写作

$$r=\frac{g(\boldsymbol{x})}{\|\boldsymbol{w}\|} \tag{5-15}$$

若 $\boldsymbol{x}$ 为原点,则

$$g(\boldsymbol{x})=w_0 \tag{5-16}$$

将式(5-16)代入式(5-15),就得到从原点到超平面 H 的距离

$$r_0=\frac{w_0}{\|\boldsymbol{w}\|} \tag{5-17}$$

如果 $w_0>0$,则原点在 H 的正侧;若 $w_0<0$,则原点在 H 的负侧。若 $w_0=0$,则 $g(\boldsymbol{x})$具有齐次形式 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$,说明超平面 H 通过原点。图 5-2 对这些结果作了几何解释。

总之,利用线性判别函数进行决策,就是用一个超平面把特征空间分割成两个决策区域。超平面的方向由权向量 $\boldsymbol{w}$ 确定,它的位置由阈值权 w_0 确定。判别函数 $g(\boldsymbol{x})$正比于 $\boldsymbol{x}$ 点到超平面的代数距离(带正负号)。当 $\boldsymbol{x}$ 在 H 正侧时,$g(\boldsymbol{x})>0$;在负侧时,$g(\boldsymbol{x})<0$。

5.4 Fisher 线性判别分析

现在从最直观的 Fisher 线性判别分析(linear discriminant analysis,LDA)开始来介绍一些最有代表性的线性判别方法。

LDA 是 R. A. Fisher 于 1936 年提出来的方法[①]。

两类的线性判别问题可以看作是把所有样本都投影到一个方向上,然后在这个一维空间中确定一个分类的阈值。过这个阈值点且与投影方向垂直的超平面就是两类的分类面。

那么,如何确定投影方向呢?

在图 5-3 的例子中,可以看到,按左图中的方向投影后两类样本可以比较好地分开,而按右图的方向投影后则两类样本混在一起。显然,左图的投影方向是更好的选择。Fisher 线性判别的思想就是,选择投影方向,使投影后两类相隔尽可能远,而同时每一类内部的样本又尽可能聚集。

为了定量地研究这一问题,我们先来定义一些基本概念。

这里只讨论两类分类的问题。训练样本集是 $\mathcal{X}=\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_N\}$,每个样本是一个 d 维向量,其中 ω_1 类的样本是 $\mathcal{X}_1=\{\boldsymbol{x}_1^1,\cdots,\boldsymbol{x}_{N_1}^1\}$,$\omega_2$ 类的样本是 $\mathcal{X}_2=\{\boldsymbol{x}_1^2,\cdots,\boldsymbol{x}_{N_2}^2\}$。我们要寻找一个投影方向 $\boldsymbol{w}$($\boldsymbol{w}$ 也是一个 d 维向量),投影以后的样本变成

---

① Fisher R A. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7: 179-188, 1936.

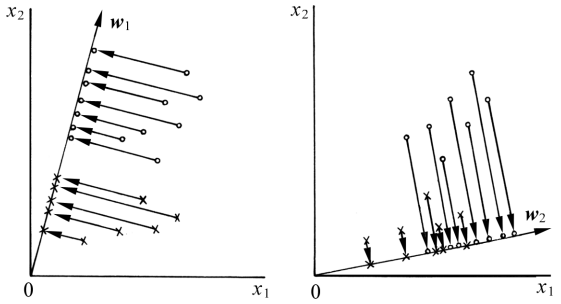


图 5-3 寻找有利于分类的投影方向

$$y_i = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i, \quad i = 1,2,\cdots,N \tag{5-18}$$

在原样本空间中,类均值向量为

$$\boldsymbol{m}_i = \frac{1}{N_i}\sum_{\boldsymbol{x}_j \in \mathscr{X}_i} \boldsymbol{x}_j, \quad i = 1,2 \tag{5-19}$$

定义各类的类内离散度矩阵(within-class scatter matrix)为

$$\boldsymbol{S}_i = \sum_{\boldsymbol{x}_j \in \mathscr{X}_i} (\boldsymbol{x}_j - \boldsymbol{m}_i)(\boldsymbol{x}_j - \boldsymbol{m}_i)^{\mathrm{T}}, \quad i = 1,2 \tag{5-20}$$

总类内离散度矩阵(pooled within-class scatter matrix)为①

$$\boldsymbol{S}_{\mathrm{w}} = \boldsymbol{S}_1 + \boldsymbol{S}_2 \tag{5-22}$$

类间离散度矩阵(between-class scatter matrix)定义为

$$\boldsymbol{S}_{\mathrm{b}} = (\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^{\mathrm{T}} \tag{5-23}$$

在投影以后的一维空间,两类的均值分别为

$$\tilde{m}_i = \frac{1}{N_i}\sum_{y_j \in \mathscr{Y}_i} y_j = \frac{1}{N_i}\sum_{\boldsymbol{x}_j \in \mathscr{X}_i} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}_i, \quad i = 1,2 \tag{5-24}$$

类内离散度不再是一个矩阵,而是一个值

$$\tilde{S}_i^2 = \sum_{y_j \in \mathscr{Y}_i} (y_j - \tilde{m}_i)^2, \quad i = 1,2 \tag{5-25}$$

总类内离散度为

$$\tilde{S}_{\mathrm{w}} = \tilde{S}_1^2 + \tilde{S}_2^2 \tag{5-26}$$

而类间离散度就成为两类均值差的平方

$$\tilde{S}_{\mathrm{b}} = (\tilde{m}_1 - \tilde{m}_2)^2 \tag{5-27}$$

前面已经提出,希望寻找的投影方向使投影以后两类尽可能分开,而各类内部又尽可能聚集,这一目标可以表示成如下的准则

$$\max J_{\mathrm{F}}(w) = \frac{\tilde{S}_{\mathrm{b}}}{\tilde{S}_{\mathrm{w}}} = \frac{(\tilde{m}_1 - \tilde{m}_2)^2}{\tilde{S}_1^2 + \tilde{S}_2^2} \tag{5-28}$$

这就是 Fisher 准则函数(Fisher's Criterion)。

---

① 有文献采用如下定义:

$$\boldsymbol{S}_i = \frac{1}{N_i}\sum_{\boldsymbol{x}_j \in \mathscr{X}_i} (\boldsymbol{x}_j - \boldsymbol{m}_i)(\boldsymbol{x}_j - \boldsymbol{m}_i)^{\mathrm{T}}, \quad i = 1,2 \tag{5-21a}$$

$$\boldsymbol{S}_{\mathrm{w}} = \frac{N_1}{N}\boldsymbol{S}_i + \frac{N_2}{N}\boldsymbol{S}_2 \tag{5-21b}$$

把式(5-18)代入式(5-27)和式(5-25),得到

$$\begin{aligned}\widetilde{S}_{\mathrm{b}} &= (\widetilde{m}_1 - \widetilde{m}_2)^2 \\ &= (\boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}_1 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}_2)^2 \\ &= \boldsymbol{w}^{\mathrm{T}}(\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^{\mathrm{T}}\boldsymbol{w} \\ &= \boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{b}}\boldsymbol{w}\end{aligned} \tag{5-29}$$

以及

$$\begin{aligned}\widetilde{S}_{\mathrm{w}} &= \widetilde{S}_1^2 + \widetilde{S}_2^2 \\ &= \sum_{\boldsymbol{x}_j \in \mathscr{X}_1} (\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}_1)^2 + \sum_{\boldsymbol{x}_j \in \mathscr{X}_2} (\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}_2)^2 \\ &= \sum_{\boldsymbol{x}_j \in \mathscr{X}_1} \boldsymbol{w}^{\mathrm{T}}(\boldsymbol{x}_j - \boldsymbol{m}_1)(\boldsymbol{x}_j - \boldsymbol{m}_1)^{\mathrm{T}}\boldsymbol{w} + \sum_{\boldsymbol{x}_j \in \mathscr{X}_2} \boldsymbol{w}^{\mathrm{T}}(\boldsymbol{x}_j - \boldsymbol{m}_2)(\boldsymbol{x}_j - \boldsymbol{m}_2)^{\mathrm{T}}\boldsymbol{w} \\ &= \boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_1\boldsymbol{w} + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_2\boldsymbol{w} \\ &= \boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w}\end{aligned} \tag{5-30}$$

因此,Fisher 判别准则变成

$$\max_{\boldsymbol{w}} J_{\mathrm{F}}(\boldsymbol{w}) = \frac{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{b}}\boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w}} \tag{5-31}$$

这一表达式在数学物理中被称作广义 Rayleigh 商(generalized Rayleigh quotient)。

应注意到,我们的目的是求使得式(5-31)最大的投影方向 $\boldsymbol{w}$。由于对 $\boldsymbol{w}$ 幅值的调节并不会影响 $\boldsymbol{w}$ 的方向,即不会影响 $J_{\mathrm{F}}(\boldsymbol{w})$的值,因此,可以设定式(5-31)的分母为非零常数而最大化分子部分,即把式(5-31)的优化问题转化为

$$\begin{aligned}&\max \quad \boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{b}}\boldsymbol{w} \\ &\text{s. t.} \quad \boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w} = c \neq 0\end{aligned} \tag{5-32}$$

其中,"s. t."表示优化问题中需要满足的约束条件(英文"subject to"的缩写)。

这是一个等式约束下的极值问题,可以通过引入拉格朗日(Lagrange)乘子转化成以下拉格朗日函数的无约束极值问题:

$$L(\boldsymbol{w},\lambda) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{b}}\boldsymbol{w} - \lambda(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w} - c) \tag{5-33}$$

在式(5-33)的极值处,应该满足

$$\frac{\partial L(\boldsymbol{w},\lambda)}{\partial \boldsymbol{w}} = 0 \tag{5-34}$$

由此可得,极值解 $\boldsymbol{w}^*$ 应满足

$$\boldsymbol{S}_{\mathrm{b}}\boldsymbol{w}^* - \lambda\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w}^* = 0 \tag{5-35}$$

假定 $\boldsymbol{S}_{\mathrm{w}}$ 是非奇异的(样本数大于维数时通常是非奇异的),可以得到

$$\boldsymbol{S}_{\mathrm{w}}^{-1}\boldsymbol{S}_{\mathrm{b}}\boldsymbol{w}^* = \lambda\boldsymbol{w}^* \tag{5-36}$$

也就是说,$\boldsymbol{w}^*$ 是矩阵 $\boldsymbol{S}_{\mathrm{w}}^{-1}\boldsymbol{S}_{\mathrm{b}}$ 的本征向量。我们把式(5-23)的 $\boldsymbol{S}_{\mathrm{b}}$ 代入,式(5-36)变成

$$\lambda\boldsymbol{w}^* = \boldsymbol{S}_{\mathrm{w}}^{-1}(\boldsymbol{m}_1 - \boldsymbol{m}_2)(\boldsymbol{m}_1 - \boldsymbol{m}_2)^{\mathrm{T}}\boldsymbol{w}^* \tag{5-37}$$

应注意到,$(\boldsymbol{m}_1 - \boldsymbol{m}_2)^{\mathrm{T}}\boldsymbol{w}^*$ 是标量,不影响 $\boldsymbol{w}^*$ 的方向,因此可以得到 $\boldsymbol{w}^*$ 的方向是由 $\boldsymbol{S}_{\mathrm{w}}^{-1}(\boldsymbol{m}_1 - \boldsymbol{m}_2)$决定的。由于我们只关心 $\boldsymbol{w}^*$ 的方向,因此可以取

$$\boldsymbol{w}^* = \boldsymbol{S}_{\mathrm{w}}^{-1}(\boldsymbol{m}_1 - \boldsymbol{m}_2) \tag{5-38}$$

这就是 Fisher 判别准则下的最优投影方向。

Fisher 线性判别投影方向也可以直接用下面另一种方法求得。

式(5-31)的解满足如下极值条件

$$\frac{\partial J_{\mathrm{F}}(\boldsymbol{w})}{\partial \boldsymbol{w}}=\boldsymbol{0} \tag{5-39}$$

将 $J_{\mathrm{F}}(\boldsymbol{w})$ 对 $\boldsymbol{w}$ 求导,可得

$$\frac{\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{m}_1-\boldsymbol{m}_2)}{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w}}\left[2(\boldsymbol{m}_1-\boldsymbol{m}_2)-2\left(\frac{\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{m}_1-\boldsymbol{m}_2)}{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w}}\right)\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w}\right]=\boldsymbol{0} \tag{5-40}$$

(此公式的推导读者可作为习题练习)。分析式(5-40),可以看到,由于 $\dfrac{\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{m}_1-\boldsymbol{m}_2)}{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{S}_{\mathrm{w}}\boldsymbol{w}}$ 是标量,在 $\boldsymbol{S}_{\mathrm{w}}$ 非奇异的条件下,式(5-40)的解满足

$$\boldsymbol{w}^{*}\ \propto \boldsymbol{S}_{\mathrm{w}}^{-1}(\boldsymbol{m}_1-\boldsymbol{m}_2) \tag{5-41}$$

由于我们只关心 $\boldsymbol{w}$ 的方向,所以式(5-38)就是式(5-40)的解。

需要注意的是,Fisher 判别函数最优的解本身只是给出了一个投影方向,并没有给出我们所要的分类面。要得到分类面,需要在投影后的方向(一维空间)上确定一个分类阈值 w_0,并采取决策规则

$$\text{若 } g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+w_0\gtrless 0,\text{则 } \boldsymbol{x}\in\begin{cases}\omega_1\\ \omega_2\end{cases} \tag{5-42}$$

回顾第 2 章中曾经讲到的,当样本是正态分布且两类协方差矩阵相同时,最优贝叶斯分类器是线性函数 $g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+w_0$,且其中

$$\boldsymbol{w}=\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2) \tag{5-43}$$

$$w_0=-\frac{1}{2}(\boldsymbol{\mu}_1+\boldsymbol{\mu}_2)^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1-\boldsymbol{\mu}_2)-\ln\frac{P(w_2)}{P(w_1)} \tag{5-44}$$

比较式(5-38)与式(5-43)可以看到,在样本为正态分布且两类协方差相同的情况下,如果把样本的算术平均作为均值的估计、把样本协方差矩阵当作是真实协方差矩阵的估计,则 Fisher 线性判别所得的方向实际就是最优贝叶斯决策的方向,因此,可以用式(5-44)来作为分类阈值,其中用 $\boldsymbol{m}_i$ 代替$\boldsymbol{\mu}_i$,用 $\boldsymbol{S}_{\mathrm{w}}^{-1}$ 代替$\boldsymbol{\Sigma}^{-1}$(采用式(5-21a)和式(5-21b)的定义),即

$$w_0=-\frac{1}{2}(\boldsymbol{m}_1+\boldsymbol{m}_2)^{\mathrm{T}}\boldsymbol{S}_{\mathrm{w}}^{-1}(\boldsymbol{m}_1-\boldsymbol{m}_2)-\ln\frac{P(w_2)}{P(w_1)} \tag{5-45}$$

在样本不是正态分布时,这种投影方向和阈值并不能保证是最优的,但通常仍可以取得较好的分类结果。

如果不考虑先验概率的不同,则可以采用阈值

$$w_0=-\frac{1}{2}(\tilde{m}_1+\tilde{m}_2) \tag{5-46}$$

或者

$$w_0=-\tilde{m} \tag{5-47}$$

其中,$\tilde{m}$ 是所有样本在投影后的均值。

把式(5-45)代入式(5-42)中并考虑到式(5-38),可以把决策规则写成

$$\text{若 } g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\left[\boldsymbol{x}-\frac{1}{2}(\boldsymbol{m}_1+\boldsymbol{m}_2)\right]\gtrless\log\frac{P(\omega_2)}{P(\omega_1)},\text{则 } \boldsymbol{x}\in\begin{cases}\omega_1\\ \omega_2\end{cases} \tag{5-48}$$

其直观的解释就是，把待决策的样本投影到Fisher判别的方向上，通过与两类均值投影的平分点相比较做出分类决策。在先验概率相同的情况下，以该平分点为两类的分界点；在先验概率不同时，分界点向先验概率小的一侧偏移，如图5-4所示。

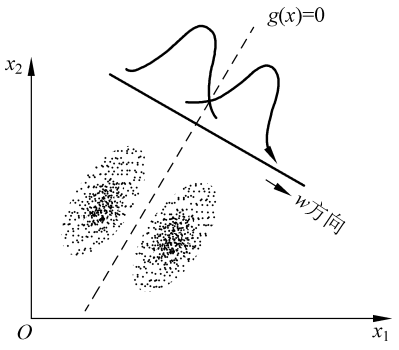


图5-4 Fisher线性判别示意图

Fisher线性判别并不对样本的分布作任何假设。但在很多情况下，当样本维数比较高且样本数也比较多时，投影到一维空间后样本接近正态分布。这时可以在一维空间中用样本拟合正态分布，用得到的参数来确定分类阈值。

5.5 感知器

Fisher线性判别是把线性分类器的设计分为两步，一是确定最优的方向，二是在这个方向上确定分类阈值。下面研究一种直接得到完整的线性判别函数 $g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+w_0$ 的方法——感知器(perceptron)。

感知器是人们设计的第一个具有学习能力的机器[①]，在机器学习和模式识别历史上扮演了重要的角色。在后面章节的介绍中我们会看到，它是多层感知器神经网络方法和各种深度学习方法的基础，也是支持向量机方法的基础。

为了讨论方便，把向量 $\boldsymbol{x}$ 增加一维，但其取值为常数，即定义

$$\boldsymbol{y}=[1,x_1,x_2,\cdots,x_d]^{\mathrm{T}} \tag{5-49}$$

其中，x_i 为样本 $\boldsymbol{x}$ 的第 i 维分量。我们称 $\boldsymbol{y}$ 为增广的样本向量。相应地，定义增广的权向量为

$$\boldsymbol{\alpha}=[w_0,w_1,w_2,\cdots,w_d]^{\mathrm{T}} \tag{5-50}$$

线性判别函数变为

$$g(\boldsymbol{y})=\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y} \tag{5-51}$$

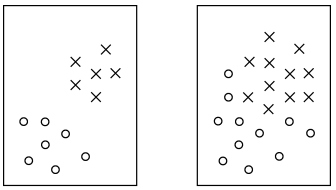
决策规则是：如果 $g(\boldsymbol{y})>0$，则 $\boldsymbol{y}\in\boldsymbol{\omega}_1$；如果 $g(\boldsymbol{y})<0$，则 $\boldsymbol{y}\in\boldsymbol{\omega}_2$。

下面定义样本集可分性的概念。

对于一组样本 $\boldsymbol{y}_1,\cdots,\boldsymbol{y}_N$，如果存在这样的权向量$\boldsymbol{\alpha}$，使得对于样本集中的任一个样本 $\boldsymbol{y}_i$，$i=1,\cdots,N$，若 $\boldsymbol{y}\in\boldsymbol{\omega}_1$ 则$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i>0$，若 $\boldsymbol{y}\in\boldsymbol{\omega}_2$ 则$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i<0$，那么称这组样本或这个样本集是线性可分的。即在样本的特征空间中，至少存在一个线性分类面能够把两类样本没有错误地分开，如图5-5(a)所示，而5-5(b)中的一组样本则是线性不可分的。

如果定义一个新的变量 $\boldsymbol{y}'$，使对于第一类的样本$\boldsymbol{y}'=\boldsymbol{y}$，而对第二类样本则 $\boldsymbol{y}'=-\boldsymbol{y}$，即

① Frank Rosenblatt, *The Perceptron-a perceiving and recognizing automaton*, Report 85-460-1, Cornell Aeronautical Laboratory, Jan. 1957.



(a) (b)

图 5-5 线性可分的样本集和线性不可分的样本集示例

$$\boldsymbol{y}'_i=\begin{cases}\boldsymbol{y}_i, & \text{若 } \boldsymbol{y}_i\in\omega_1\\ -\boldsymbol{y}_i, & \text{若 } \boldsymbol{y}_i\in\omega_2\end{cases}\qquad i=1,2,\cdots,N \tag{5-52}$$

则样本可分性条件就变成了存在$\boldsymbol{\alpha}$，使

$$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}'_i>0,\quad i=1,2,\cdots,N \tag{5-53}$$

这样定义的$\boldsymbol{y}'$称作规范化增广样本向量。在本节和下一节，为了讨论方便，都采用规范化增广样本向量，并且把$\boldsymbol{y}'$仍然记作$\boldsymbol{y}$。

本小节只讨论样本线性可分的情况。

对于线性可分的一组样本$\boldsymbol{y}_1,\cdots,\boldsymbol{y}_N$（采用规范化增广样本向量表示），如果一个权向量$\boldsymbol{\alpha}^*$满足

$$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i>0,\quad i=1,2,\cdots,N \tag{5-54}$$

则称$\boldsymbol{\alpha}^*$为一个解向量。在权值空间中所有解向量组成的区域称作解区。

显然，权向量和样本向量的维数相同，可以把权向量画到样本空间。对于一个样本$\boldsymbol{y}_i$，$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i=0$定义了权空间中一个过原点的超平面$\hat{H}_i$。对于这个样本来说，处于超平面$\hat{H}_i$正侧的任何一个向量都能使$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i>0$，因而都是对这个样本的一个解。考虑样本集中的所有样本，解区就是每个样本对应超平面的正侧的交集，如图 5-6 所示。

解区中的任意一个向量都是解向量，都能把样本没有错误地分开。但是，从直观角度看，如果一个解向量靠近解区的边缘，虽然所有样本都能满足$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i>0$，但某些样本的判别函数可能刚刚大于零，考虑到噪声、数值计算误差等因素，靠近解区中间的解向量应该更加可靠。因此，人们提出了余量的概念，即把解区向中间缩小，不取靠近边缘的解，如图 5-7 所示。形式化表示就是，引入余量$b>0$，要求解向量对满足

$$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i>b,\quad i=1,2,\cdots,N \tag{5-55}$$

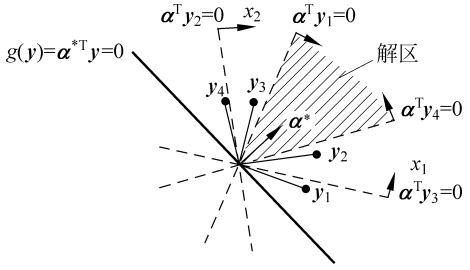


图 5-6 解向量和解区

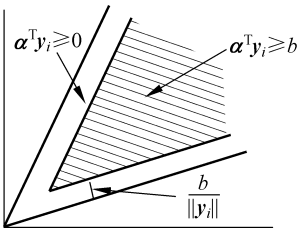


图 5-7 带有余量的解区

下面我们来看如何找到一个解向量。

对于权向量$\boldsymbol{\alpha}$，如果某个样本$\boldsymbol{y}_k$被错误分类，则$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_k\leqslant 0$。我们可以用对所有错分样本的求和来表示对错分样本的惩罚

$$J_{\mathrm{P}}(\boldsymbol{\alpha})=\sum_{\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_k\leqslant 0}(-\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_k) \tag{5-56}$$

这就是20世纪50年代Rosenblatt提出的感知器(Perceptron)准则函数[①]。

显然,当且仅当 $J_p(\boldsymbol{\alpha}^*)=\min J_P(\boldsymbol{\alpha})=0$ 时 $\boldsymbol{\alpha}^*$ 是解向量。

感知器准则函数式(5-46)的最小化可以用梯度下降方法迭代求解

$$\boldsymbol{\alpha}(t+1)=\boldsymbol{\alpha}(t)-\rho_t \nabla J_P(\boldsymbol{\alpha}) \tag{5-57}$$

即下一时刻的权向量是把当前时刻的权向量向目标函数的负梯度方向调整一个修正量,其中 ρ_t 为调整的步长。目标函数 J_P 对权向量 $\boldsymbol{\alpha}$ 的梯度是

$$\nabla J_P(\boldsymbol{\alpha})=\frac{\partial J_P(\boldsymbol{\alpha})}{\partial \boldsymbol{\alpha}}=\sum_{\boldsymbol{\alpha}^T\boldsymbol{y}_k\leqslant 0}(-\boldsymbol{y}_k) \tag{5-58}$$

因此,迭代修正的公式就是

$$\boldsymbol{\alpha}(t+1)=\boldsymbol{\alpha}(t)+\rho_t\sum_{\boldsymbol{\alpha}^T\boldsymbol{y}_k\leqslant 0}\boldsymbol{y}_k \tag{5-59}$$

即在每一步迭代时把错分的样本按照某个系数加到权向量上。

通常情况下,一次将所有错误样本都进行修正的做法并不是效率最高的,更常用的是每次只修正一个样本的固定增量法,算法步骤是:

(1) 任意选择初始的权向量 $\boldsymbol{\alpha}(0)$,置 $t=0$;

(2) 考查样本 $\boldsymbol{y}_j$,若 $\boldsymbol{\alpha}(t)^T\boldsymbol{y}_j\leqslant 0$,则 $\boldsymbol{\alpha}(t+1)=\boldsymbol{\alpha}(t)+\boldsymbol{y}_j$,否则继续;

(3) 考查另一个样本,重复(2),直至对所有样本都有 $\boldsymbol{\alpha}(t)^T\boldsymbol{y}_j>0$,即 $J_P(\boldsymbol{\alpha})=0$。

如果考虑余量 b,则只需将上面的算法中的错分判断条件变成 $\boldsymbol{\alpha}(t)^T\boldsymbol{y}_j\leqslant b$ 即可。

可以证明,对于线性可分的样本集,采用这种梯度下降的迭代算法,经过有限次修正后一定会收敛到一个解向量 $\boldsymbol{\alpha}^*$。这里不给出严格的证明,而是用图5-8的例子来直观地说明这一收敛过程。

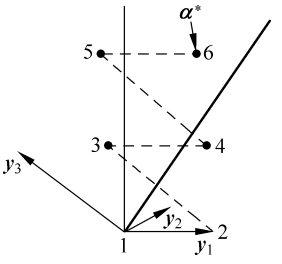


图5-8 感知器学习算法收敛过程示意

在图5-8的例子中,只有三个样本 $\boldsymbol{y}_1,\boldsymbol{y}_2,\boldsymbol{y}_3$(注意是规范化增广样本向量)。假设令权向量初值为 $\boldsymbol{\alpha}(0)=\boldsymbol{0}$(零向量,图5-8中的"1"位置);在第一步,考查 $\boldsymbol{y}_1$,$\boldsymbol{\alpha}(0)^T\boldsymbol{y}_1=0$,所以需要向 $\boldsymbol{y}_1$ 的方向修正权值 $\boldsymbol{\alpha}(1)=\boldsymbol{\alpha}(0)+\boldsymbol{y}_1$,权向量变成图5-8中的第2个点;下一步,考查 $\boldsymbol{y}_2$,$\boldsymbol{\alpha}(1)^T\boldsymbol{y}_2>0$,再考查 $\boldsymbol{y}_3$,发现 $\boldsymbol{\alpha}(1)^T\boldsymbol{y}_3<0$,所以采取修正 $\boldsymbol{\alpha}(2)=\boldsymbol{\alpha}(1)+\boldsymbol{y}_3$,得到了图中的第3个点;第四步发现 $\boldsymbol{y}_1$ 又被分错,$\boldsymbol{\alpha}(2)^T\boldsymbol{y}_1<0$,再采取修正 $\boldsymbol{\alpha}(3)=\boldsymbol{\alpha}(2)+\boldsymbol{y}_1$,权向量变成了图中的第4个点;第五步,$\boldsymbol{y}_2$ 依然是分类正确的,而 $\boldsymbol{y}_3$ 又被分错,所以需再次向 $\boldsymbol{y}_3$ 方向调整权值 $\boldsymbol{\alpha}(4)=\boldsymbol{\alpha}(3)+\boldsymbol{y}_3$,变成了图中的第5个点;第六步,由于新的权值又对 $\boldsymbol{y}_1$ 错分了,所以再次将 $\boldsymbol{y}_1$ 方向调整,$\boldsymbol{\alpha}(5)=\boldsymbol{\alpha}(4)+\boldsymbol{y}_1$,得到第6个点所示的权向量。此时再次考查3个训练样本,发现都被正确分类了,$\boldsymbol{\alpha}(5)$就是迭代求得的解向量。不难想象,不论样本数目和维数如何,只要解区存在(样本线性可分),那么根据相同的原理,总可以经过有限步的迭代求得一个解向量。

这种单步的固定增量法采用的修正步长是 $\rho_t=1$。为了减少迭代步数,人们还提出可

① Rosenblatt F. The perceptron: a probabilistic model for information storage and organization in the brain. Cornell Aeronautical Laboratory, *Psychological Review*, 1958, 65(6): 386-408.

以使用可变的步长，例如绝对修正法就是对错分样本 $\boldsymbol{y}_j$ 用下面的步长来调整权向量

$$\rho_t = \frac{|\boldsymbol{\alpha}(k)^{\mathrm{T}}\boldsymbol{y}_j|}{\|\boldsymbol{y}_j\|^2} \tag{5-60}$$

感知器算法是最简单的可以学习的机器。由于它只能解决线性可分的问题，所以，在实际应用中，直接使用感知器算法的场合并不多。但是，它是很多更复杂的算法的基础，例如第 6 章将要介绍的支持向量机和多层感知器人工神经网络。

在感知器准则中，要求全部样本是线性可分的。此时，经过有限步的迭代梯度下降法就可以收敛到一个解。当样本不是线性可分时，如果仍然使用感知器算法，则算法不会收敛。如果任意地让算法停止在某一时刻，则无法保证得到的解是有用的(能够把较多的样本正确分类)。人们研究了很多策略来设法使感知器算法在样本集不是线性可分时仍能得到合理有用的解，其中一种比较常用的做法是，在梯度下降过程中让步长按照一定的启发式规则逐渐缩小，这样就可以强制算法收敛，而且往往可以得到有用的解。如果多数样本是可分的，那么这种简单的做法在很多情况下还是有效的。

5.6 最小平方误差判别

这一节讨论考虑线性不可分样本集的分类方法。在线性不可分的情况下，不等式组

$$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i > 0, \quad i = 1,2,\cdots,N \tag{5-61}$$

不可能同时满足。一种直观的想法就是，希望求解一个$\boldsymbol{\alpha}^*$使被错分的样本尽可能少，即不满足不等式(5-61)的样本尽可能少，这种方法是通过解线性不等式组来最小化错分样本数目，通常采用搜索算法求解。

但是，求解线性不等式组有时并不方便，为了避免此问题，可以引进一系列待定的常数，把不等式组(5-61)转变成下列方程组

$$\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{y}_i = b_i > 0, \quad i = 1,2,\cdots,N \tag{5-62}$$

或写成矩阵形式

$$\boldsymbol{Y}\boldsymbol{\alpha} = \boldsymbol{b} \tag{5-63}$$

其中

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{y}_1^{\mathrm{T}} \\ \vdots \\ \boldsymbol{y}_N^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} y_{11} & \cdots & y_{1\hat{d}} \\ \vdots & \ddots & \vdots \\ y_{N1} & \cdots & y_{N\hat{d}} \end{bmatrix} \tag{5-64}$$

$$\boldsymbol{b} = [b_1, b_2, \cdots, b_N]^{\mathrm{T}} \tag{5-65}$$

其中 $\hat{d}$ 是增广的样本向量的维数，$\hat{d} = d + 1$。暂且不考虑常数向量 $\boldsymbol{b}$ 如何确定的问题，先来看这个方程组的求解。

很显然，这个方程组求解的问题就是 5.2 节中讨论的线性回归问题，只不过这里的响应变量是人为对每个样本给定的 b_i。

通常情况下，$N > \hat{d}$，所以式(5-63)中的方程个数大于未知数个数，属于矛盾方程组，无法求得精确解。方程组的误差为 $\boldsymbol{e} = \boldsymbol{Y}\boldsymbol{\alpha} - \boldsymbol{b}$，可以求解方程组的最小平方误差解，即

$$\boldsymbol{\alpha}^* : \quad \min_{\boldsymbol{\alpha}} J_S(\boldsymbol{\alpha}) \tag{5-66}$$

其中 $J_S(\boldsymbol{\alpha})$ 是最小平方误差(MSE)准则函数

$$J_S(\boldsymbol{\alpha}) = \|\boldsymbol{Y}\boldsymbol{\alpha} - \boldsymbol{b}\|^2 = \sum_{i=1}^{N} (\boldsymbol{\alpha}^T \boldsymbol{y}_i - b_i)^2 \tag{5-67}$$

这个准则函数的最小化主要有两类方法：伪逆法求解与梯度下降法求解。

$J_S(\boldsymbol{\alpha})$ 在极值处对 $\boldsymbol{\alpha}$ 的梯度应该为零，依此可以得到

$$\nabla J_S(\boldsymbol{\alpha}) = 2\boldsymbol{Y}^T(\boldsymbol{Y}\boldsymbol{\alpha} - \boldsymbol{b}) = 0 \tag{5-68}$$

可得

$$\boldsymbol{\alpha}^* = (\boldsymbol{Y}^T\boldsymbol{Y})^{-1}\boldsymbol{Y}^T\boldsymbol{b} = \boldsymbol{Y}^+\boldsymbol{b} \tag{5-69}$$

其中 $\boldsymbol{Y}^+ = (\boldsymbol{Y}^T\boldsymbol{Y})^{-1}\boldsymbol{Y}^T$ 是长方矩阵 $\boldsymbol{Y}$ 的伪逆。

也可以用梯度下降法来迭代求解式(5-67)的最小值。算法如下：

(1) 任意选择初始的权向量 $\boldsymbol{\alpha}(0)$，置 $t=0$；

(2) 按照梯度下降的方向迭代更新权向量

$$\boldsymbol{\alpha}(t+1) = \boldsymbol{\alpha}(t) - \rho_t \boldsymbol{Y}^T(\boldsymbol{Y}\boldsymbol{\alpha} - \boldsymbol{b}) \tag{5-70}$$

直到满足 $\nabla J_S(\boldsymbol{\alpha}) \leqslant \xi$ 或者 $\|\boldsymbol{\alpha}(t+1) - \boldsymbol{\alpha}(t)\| \leqslant \xi$ 时为止，其中 ξ 是事先确定的误差灵敏度。

参照感知器算法中的单步修正法，对最小平方误差准则，也可以采用单样本修正法来调整权向量

$$\boldsymbol{\alpha}(t+1) = \boldsymbol{\alpha}(t) + \rho_t(b_k - \boldsymbol{\alpha}(t)^T\boldsymbol{y}_k)\boldsymbol{y}_k \tag{5-71}$$

其中，$\boldsymbol{y}_k$ 是使得 $\boldsymbol{\alpha}(t)^T\boldsymbol{y}_k \neq b_k$ 的样本。

这种算法称作 Widrow-Hoff 算法，也称作最小均方根算法或 LMS 算法(least-mean-square algorithm)。历史上，用这种算法构造的学习机器称作 ADALINE[①]，与感知器一起是现在神经网络类学习机器的最早形式。

显然，我们也同样可以用类似式(5-71)的迭代算法来实现 5.2 节中提出的线性回归问题的迭代求解。

上面一直没有讨论 $\boldsymbol{b}$ 的选取问题。选择不同的 $\boldsymbol{b}$ 会带来不同的结果。可以证明，如果对应同一类样本的 b_i 选择为相同的值，那么最小平方误差方法的解等价于 Fisher 线性判别的解，把样本和权向量都还原成增广以前的形式后有

$$\boldsymbol{w}^* \propto \boldsymbol{S}_W^{-1}(\boldsymbol{m}_1 - \boldsymbol{m}_2) \tag{5-72}$$

其中，$\boldsymbol{m}_1$、$\boldsymbol{m}_2$ 是两类各自的均值向量，$\boldsymbol{S}_W$ 是总类内离散度矩阵。特别地，当 $\boldsymbol{b}$ 的选择为第一类样本对应的 b_i 都是 N/N_1，第二类样本对应的 b_i 都是 N/N_2 时，阈值 w_0^* 为样本均值在所得一维判别函数方向的投影，即

$$w_0 = -\boldsymbol{m}^T\boldsymbol{w}^* \tag{5-73}$$

其中，N_1、N_2 分别是第一类和第二类的样本数，N 是样本总数，$\boldsymbol{m}$ 是全部样本的均值，即 $\boldsymbol{m} = \frac{1}{N}(N_1\boldsymbol{m}_1 + N_2\boldsymbol{m}_2)$。

另外还可以证明，如果对所有样本都取 $b_i=1$，那么当 $N \to \infty$ 时，MSE 算法的解是贝叶

---

① Widrow & Hoff, Adaptive switching circuits, 1960 *IRE Western Electric Show and Convention Record*, *Part* 4, pp. 96-104, Aug, 1960.

斯判别函数

$$g_0(\boldsymbol{x}) = P(\omega_1 \mid \boldsymbol{x}) - P(\omega_2 \mid \boldsymbol{x}) \tag{5-74}$$

的最小平方误差逼近。即，下面定义的均方逼近误差

$$\varepsilon^2 = \int [\boldsymbol{\alpha}^{\mathrm{T}} \boldsymbol{y} - g_0(\boldsymbol{x})]^2 p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \tag{5-75}$$

在 $\boldsymbol{\alpha}^* = \boldsymbol{Y}^+ \mathbf{1}_N$ 时取得最小值，其中 $\mathbf{1}_N$ 表示由 N 个 1 组成的列向量。

5.7 罗杰斯特回归[①]

在 5.2 节中简要介绍的线性回归，在很多实际问题中有大量应用，例如我们可以用它来研究身高与体重的关系。事实上，推动线性回归走入应用的数学家 Adolphe Quetelet 基于身高与体重关系所定义的体重指数 BMI 一直被沿用至今。图 5-9 给出了一组样本上得到的人的血压(收缩压)与年龄的关系，这种线性回归对我们从数据中认识规律发挥了重要作用，从数据中学习到的线性模型也可以用来对连续取值的响应变量进行定量预测。

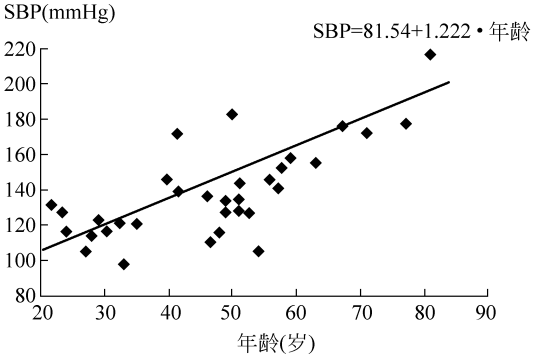


图 5-9 线性回归的例子(Colton T. *Statistics in Medicine*. Boston: Little Brown, 1974)

一般情况下，所关心的变量可能与多个自变量有关系，这就是多元线性回归问题，即求下列线性模型中的系数的问题：$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m + \varepsilon$，其中，$y$ 是我们要回归的变量，$x_1, \cdots, x_m$ 是与它有关系的特征变量，$\beta_1, \cdots, \beta_m$ 是它们对应的系数，β_0 是常数项，ε 是回归的残差(亦称离差)，即用 $\boldsymbol{x}$ 的线性函数 $\beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m$ 估计 y 带来的误差。特征 x_i 可以是连续变量，也可以是离散变量，如性别、基因型等。

求解线性回归的最基本方法就是最小二乘法，即求使各样本残差的平方和达到最小的

---

① 罗杰斯特回归(Logistic regression)：国内部分文献和教科书译为“逻辑回归”，个别在线词典也把 logistic 翻译为“逻辑的”，笔者经考证后认为不妥。从语言学上 logistic 一词与 logic 并无关系，故笔者采用了部分统计学文献中音译的做法，类似的音译还有“罗杰斯蒂回归”等。Logistic 这个形容词来源于名词 logistics，据牛津字典，其含义是 The detailed coordination of a complex operation involving many people, facilities, or supplies，即涉及很多人、设施和物资的复杂任务的详细协调，经常用于指军事或其他重大行动中的后勤，也指现代社会中的物流。法国数学家 Pierre-Francois Verhulst(1804—1849)在 1845 年把图 5-10 中的曲线用法文称作 courbe logistique，即罗杰斯特曲线(Logistic curve)，把对应的函数称作罗杰斯特函数(logistic function)。当时他在文献中并未对名字进行解释，后人就一直沿用这个名字。

系数 β_i。这实际是在变量 y 服从正态分布假设下的最大似然估计。(可作为习题留给读者练习)

系数 β_i 的直观解释是,当其他因素都不变时,特征 x_i 增加一个单位所带来的 y 的变化。由于多元线性回归能够把特征的作用量化,在很多根据观测数据研究未知机理的问题里有很广泛的应用。

在模式识别问题中,所关心的量是分类,例如是否会患某种疾病,这时就不能再用简单的线性回归的方法来研究特征与分类之间的关系。

人们观察到,现实世界里有很多这样的情况,就是某个因素对于事物性质的影响是按比例渐进的:设 $x\in R$ 是样本的特征(例如体重指数),考查样本是否属于所关心类别(如患有某种疾病),我们很难建立患病($y=1$)或不患病($y=-1$)与 x 的关系,但却经常可以发现 x 与人群中患病概率的关系符合图 5-10 的 Logistic 函数所示的形式,即

$$P(y=1\mid x)=\frac{e^{w_0+w_1x}}{1+e^{w_0+w_1x}} \tag{5-76}$$

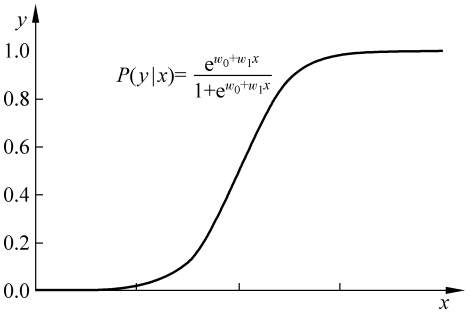


图 5-10 Logistic 函数

其中 $P(y=1|x)$经常简记作 $P(y|x)$。这种函数被称作罗杰斯特函数,在神经网络中被称作 Sigmoid 函数。人们经常用 $\theta(s)$来代表罗杰斯特函数,即

$$\theta(s)=\frac{e^s}{1+e^s}=\frac{1}{1+e^{-s}} \tag{5-77}$$

容易证明,

$$\theta(-s)=1-\theta(s) \tag{5-78}$$

在医学研究中,人们经常关心一个称作“几率”(odds)的概念,指患某种疾病的可能性与不患病的可能性之比。如果患病概率符合 Logistic 函数,则几率为

$$\frac{P(y\mid x)}{1-P(y\mid x)}=e^{w_0+w_1x} \tag{5-79}$$

对它取自然对数,得到对数几率(log odds)

$$\ln\left(\frac{P(y\mid x)}{1-P(y\mid x)}\right)=w_0+w_1x \tag{5-80}$$

公式左边被称作 $P(y|x)$的 logit 函数。变量 y 与 x 之间的这种关系模型称作 logistic 模型,其中的 w_1 反映了当 x 增加一个单位时,样本属于 $y=1$ 类的几率在对数尺度上增加的幅度。

正如线性回归中一样,logistic 模型也可以有多个自变量 $x_1,\cdots,x_m$,即样本 $\boldsymbol{x}$ 是由 m 维特征组成的,这些特征可以是连续特征,也可以是离散特征。

多元的 logit 函数是

$$\text{logit}(\boldsymbol{x})=\ln\left(\frac{P(y|\boldsymbol{x})}{1-P(y|\boldsymbol{x})}\right)=w_0+w_1x_1+\cdots+w_mx_m \tag{5-81}$$

样本属于 $y=1$ 类的概率是

$$P(y|\boldsymbol{x})=\frac{\mathrm{e}^{w_0+w_1x_1+\cdots+w_mx_m}}{1+\mathrm{e}^{w_0+w_1x_1+\cdots+w_mx_m}} \tag{5-82}$$

罗杰斯特回归(Logistic regression)就是用式(5-81)的对数几率模型,即式(5-82)的概率模型来描述样本属于某类的可能性与样本特征之间的关系,用训练数据来估计式(5-82)中的系数。得到这些系数后,罗杰斯特回归的决策函数是

$$\text{若 logit}(\boldsymbol{x})\gtrless\begin{matrix}0\\0\end{matrix},\quad \text{则}\begin{cases}\boldsymbol{x}\in\omega_1\\ \boldsymbol{x}\in\omega_2\end{cases} \tag{5-83}$$

罗杰斯特回归最基本的学习算法是最大似然法。

设共有 N 个独立的训练样本$\{(\boldsymbol{x}_1,y_1),\cdots,(\boldsymbol{x}_N,y_N)\}$,$\boldsymbol{x}_j\in R^{d+1}$,$y_j\in\{-1,1\}$,其中 $y=1$ 表示样本属于所关心类别,$y=-1$ 表示不属于。我们假设样本的类别是从某个未知的概率 $f(\boldsymbol{x})$中产生出来的,以概率 $f(\boldsymbol{x})$属于所关心类别,以概率 $1-f(\boldsymbol{x})$不属于该类,即

$$P(y\mid\boldsymbol{x})=\begin{cases}f(\boldsymbol{x}), & y=+1\\ 1-f(\boldsymbol{x}), & y=-1\end{cases} \tag{5-84}$$

用罗杰斯特函数 $h(\boldsymbol{x})=\theta(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x})$来估计 $f(\boldsymbol{x})$,其中 $\boldsymbol{w}$ 是罗杰斯特函数中待求参数组成的向量。

对于样板集中的一个样本实例$(\boldsymbol{x}_j,y_j)$,$\boldsymbol{x}_j$ 和 y_j 都是已知的,而概率模型 $h(\boldsymbol{x})$未知,于是下面的概率就度量了该已经发生的样本是从该未知的模型中产生出来的可能性:

$$P(y_j\mid\boldsymbol{x}_j)=\begin{cases}h(\boldsymbol{x}_j), & y_j=+1\\ 1-h(\boldsymbol{x}_j), & y_j=-1\end{cases} \tag{5-85}$$

称作模型 h 在样本上的似然函数。

注意到 $\theta(-s)=1-\theta(s)$,我们可以把上式的两种情况合并在一起,并记作 $l(\cdot)$,即

$$l(h\mid(\boldsymbol{x}_j,y_j))\triangleq P(y_j\mid\boldsymbol{x}_j,h)=\theta(y_j\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j) \tag{5-86}$$

它是模型 h 在样本$(\boldsymbol{x}_j,y_j)$上的似然函数。

对于样本集中所有的样本,模型的似然函数是

$$L(\boldsymbol{w})=\prod_{j=1}^{N}P(y_j\mid\boldsymbol{x}_j)=\prod_{j=1}^{N}\theta(y_j\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j) \tag{5-87}$$

这里,我们把似然函数显式地写成模型中未知参数 $\boldsymbol{w}$ 的函数。罗杰斯特回归就是要用训练样本集来估计参数 $\boldsymbol{w}$,使样本集是从这个模型中产生出来的可能性最大。

我们可以沿用在感知器和最小平方误差方法中的策略,用梯度下降的方法来最优化目标函数。为此,我们定义目标函数为似然函数的负对数,优化问题是

$$\begin{aligned}\min\quad E(\boldsymbol{w})&=-\frac{1}{N}\ln(L(\boldsymbol{w}))=-\frac{1}{N}\ln\left(\textstyle\prod_{j=1}^{N}\theta(y_j\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j)\right)\\&=\frac{1}{N}\sum_{j=1}^{N}\ln\left(\frac{1}{\theta(y_j\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j)}\right)\end{aligned}$$

$$=\frac{1}{N}\sum_{j=1}^{N}\ln(1+\mathrm{e}^{-y_j\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_j}) \tag{5-88}$$

于是,我们得到下面的采用梯度下降法寻优的罗杰斯特回归算法:

(1) 记时刻为 $k=0$,初始化参数 $\boldsymbol{w}(0)$。

(2) 计算目标函数的负梯度方向

$$\nabla E=-\frac{1}{N}\sum_{j=1}^{N}\frac{y_j\boldsymbol{x}_j}{1+\mathrm{e}^{y_j\boldsymbol{w}(k)^{\mathrm{T}}\boldsymbol{x}_j}}$$

按步长(学习率)η 更新下一时刻参数

$$\boldsymbol{w}(k+1)=\boldsymbol{w}(k)-\eta\nabla E$$

检查是否达到终止条件,如未达到,令 $k=k+1$,重新进行(2)。

(3) 算法停止,输出得到的参数 $\boldsymbol{w}$。

其中终止条件可以是似然函数的梯度已经小于某个预设值,训练过程不再有显著更新,或者是迭代达到预设的上限,等等。

5.8 最优分类超平面与线性支持向量机

现在再回到线性可分情况。容易发现,只要一个样本集线性可分,就肯定存在无数多解,解区中的任何向量都是一个解向量。感知器算法采用不同的初始值和不同的迭代参数就会得到不同的解。如图 5-11 所示,在这些解中,哪一个更好呢?

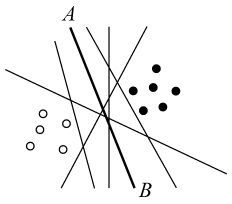


图 5-11 线性可分情况下的多解性

5.8.1 最优分类超平面

对于图 5-11 中的例子,如果要求手动画出一条分类线,多数人会倾向于画在两类的中间,大约线 A-B 的位置上,因为这条分类线离两类样本都最远。下面来形式化地定义这样的分类线(面)。这里我们使用原始的样本向量表示而不采用增广向量。

假定有训练样本集

$$(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\cdots,(\boldsymbol{x}_N,y_N),\quad \boldsymbol{x}_i\in R^d,y_i\in\{+1,-1\} \tag{5-89}$$

其中每个样本是 d 维向量,y 是类别标号,ω_1 类用 $+1$ 表示,ω_2 类用 -1 表示。这些样本是线性可分的,即存在超平面

$$g(\boldsymbol{x})=(\boldsymbol{w}\cdot\boldsymbol{x})+b=0 \tag{5-90}$$

把所有 N 个样本都没有错误地分开。这里,$\boldsymbol{w}\in R^d$ 是线性判别函数的权值,b 是其中的常

数项，在前面几小节中都用 w_0 表示，而这里为了与其他有关支持向量机的文献一致，我们用 b 来表示。$(\boldsymbol{w}\cdot\boldsymbol{x})$表示向量 $\boldsymbol{w}$ 与 $\boldsymbol{x}$ 的内积，即 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}$。

定义：一个超平面，如果它能够将训练样本没有错误地分开，并且两类训练样本中离超平面最近的样本与超平面之间的距离是最大的，则把这个超平面称作最优分类超平面(optimal separating hyperplane)，简称最优超平面(optimal hyperplane)。两类样本中离分类面最近的样本到分类面的距离称作分类间隔(margin)，最优超平面也称作最大间隔超平面，如图 5-12 所示。

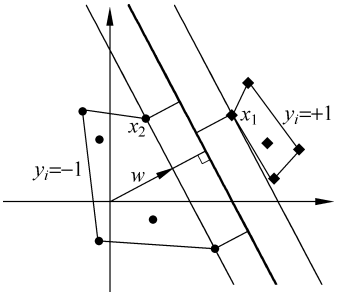


图 5-12 分类间隔与最优超平面

最优超平面定义的分类决策函数为

$$f(\boldsymbol{x})=\mathrm{sgn}(g(\boldsymbol{x}))=\mathrm{sgn}((\boldsymbol{w}\cdot\boldsymbol{x})+b) \quad (5\text{-}91)$$

其中，sgn(·)为符号函数，当自变量为正值时函数取值为 1，自变量为负值时函数取值为−1。

根据 5.3 节的基本知识我们知道，向量 $\boldsymbol{x}$ 到分类面 $g(\boldsymbol{x})=0$ 的距离是 $|g(\boldsymbol{x})|/\|\boldsymbol{w}\|$，其中 $\|\boldsymbol{w}\|$ 是权向量的模，即 $\|\boldsymbol{w}\|=(\boldsymbol{w}\cdot\boldsymbol{w})^{1/2}$。

容易注意到，对于式(5-92)的决策函数，对权值 $\boldsymbol{w}$ 和 b 作任何正的尺度调整都不会影响分类决策，同时也不会改变样本到分类面的距离，因此上面定义的最优分类面没有唯一解，而是有无数多个等价的解。为了使这一问题有唯一解，需要把 $\boldsymbol{w}$ 和 b 的尺度确定下来。

所有 N 个样本都可以被超平面没有错误地分开，就是要求所有样本都满足

$$\begin{cases}(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b>0, & y_i=+1\\ (\boldsymbol{w}\cdot\boldsymbol{x}_i)+b<0, & y_i=-1\end{cases} \quad (5\text{-}92)$$

既然尺度可以调整，我们可以把式(5-91)的条件变成

$$\begin{cases}(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b\geqslant 1, & y_i=+1\\ (\boldsymbol{w}\cdot\boldsymbol{x}_i)+b\leqslant -1, & y_i=-1\end{cases} \quad (5\text{-}93)$$

即要求第一类样本中 $g(\boldsymbol{x})$最小等于 1，而第二类样本中 $g(\boldsymbol{x})$最大等于−1。把样本的类别标号 y 值乘到不等式(5-93)中，可以把两个不等式合并成一个统一的形式：

$$y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]\geqslant 1, \quad i=1,2,\cdots,N \quad (5\text{-}94)$$

用此条件约束分类超平面的权值尺度变化，这种超平面称作规范化的分类超平面(the canonical form of the separating hyperplane)。$g(\boldsymbol{x})=1$ 和 $g(\boldsymbol{x})=-1$ 就是过两类中各自离分类面最近的样本且与分类面平行的两个边界超平面。

如图 5-13 所示，由于限制两类离分类面最近的样本的 $g(\boldsymbol{x})$分别等于 1 和−1，那么分类间隔就是 $M=\dfrac{2}{\|\boldsymbol{w}\|}$。于是，求解最优超平面的问题就成为

$$\min_{\boldsymbol{w},b} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 \quad (5\text{-}95)$$

$$\text{s.t.} \quad y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]-1\geqslant 0, \quad i=1,2,\cdots,N \quad (5\text{-}96)$$

这是一个在不等式约束下的优化问题，可以通过拉格朗日法求解。对每个样本引入一个拉格朗日系数

$$\alpha_i\geqslant 0, \quad i=1,\cdots,N \quad (5\text{-}97)$$

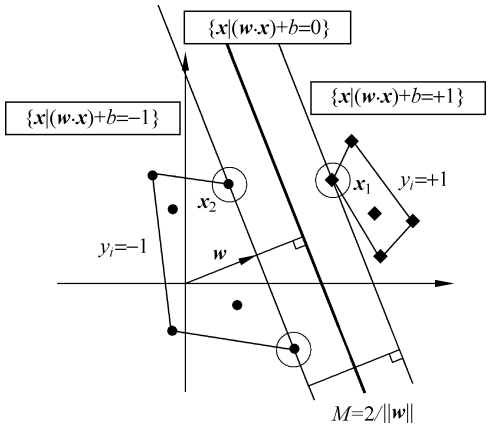


图 5-13 规范化的最优分类面

可以把式(5-72)和式(5-73)的优化问题等价地转化为下面的问题

$$\min_{\boldsymbol{w},b}\max_{\boldsymbol{\alpha}} L(\boldsymbol{w},b,\boldsymbol{\alpha}) = \frac{1}{2}(\boldsymbol{w}\cdot\boldsymbol{w}) - \sum_{i=1}^{N}\alpha_i\{y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]-1\} \tag{5-98}$$

式中的 $L(\boldsymbol{w},b,\boldsymbol{\alpha})$是拉格朗日泛函,式(5-95)、式(5-96)的解等价于式(5-98)对 $\boldsymbol{w}$ 和 b 求最小、而对$\boldsymbol{\alpha}$ 求最大,最优解在 $L(\boldsymbol{w},b,\boldsymbol{\alpha})$的鞍点上取得,如图 5-14 所示。

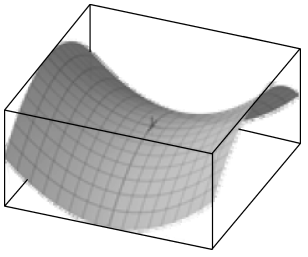


图 5-14 鞍点示意图

在式(5-98)的鞍点处,目标函数 $L(\boldsymbol{w},b,\boldsymbol{\alpha})$对 $\boldsymbol{w}$ 和 b 的偏导数都为零,由此我们可以得到,对最优解处,有

$$\boldsymbol{w}^* = \sum_{i=1}^{N}\alpha_i^* y_i \boldsymbol{x}_i \tag{5-99}$$

且

$$\sum_{i=1}^{N} y_i \alpha_i^* = 0 \tag{5-100}$$

将这两个条件代入拉格朗日泛函中可以得到,式(5-95)、式(5-96)的最优超平面问题的解等价于下面的优化问题的解

$$\max_{\boldsymbol{\alpha}} \quad Q(\boldsymbol{\alpha}) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j y_i y_j (\boldsymbol{x}_i\cdot\boldsymbol{x}_j) \tag{5-101}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} y_i\alpha_i = 0 \tag{5-102}$$

且

$$\alpha_i \geqslant 0, \quad i=1,\cdots,N \tag{5-103}$$

这是一个对 $\alpha_i, i=1,\cdots,N$ 的二次优化问题，称作最优超平面的对偶问题(the dual problem)，而式(5-95)、式(5-96)的优化问题称作最优超平面的原问题(the primary problem)。通过对偶问题的解 $\alpha_i^*, i=1,\cdots,N$，可以求出原问题的解

$$\boldsymbol{w}^* = \sum_{i=1}^{N} \alpha_i^* y_i \boldsymbol{x}_i \tag{5-104}$$

$$f(\boldsymbol{x}) = \operatorname{sgn}\{g(\boldsymbol{x})\} = \operatorname{sgn}\{(\boldsymbol{w}^* \cdot \boldsymbol{x}) + b\} = \operatorname{sgn}\{\sum_{i=1}^{N} \alpha_i^* y_i (\boldsymbol{x}_i \cdot \boldsymbol{x}) + b^*\} \tag{5-105}$$

即，最优超平面的权值向量等于训练样本以一定的系数加权后进行线性组合。

应注意到，在判别函数式(5-105)中的 b^* 尚没有得到。现在来看 b^* 的求解问题。

根据最优化理论中的库恩-塔克(Kuhn-Tucker)条件，式(5-98)中的拉格朗日泛函的鞍点处满足

$$\alpha_i \{y_i [(\boldsymbol{w} \cdot \boldsymbol{x}_i) + b] - 1\} = 0, \quad i=1,2,\cdots,N \tag{5-106}$$

再考虑到式(5-96)和式(5-97)，可以看到，对于满足式(5-96)中大于号的样本，必定有 $\alpha_i = 0$。而只有那些使式(5-96)中等号成立的样本所对应的 α_i 才会大于 0。这些样本就是离分类面最近的那些样本，从图 5-13 中可以看到，是这些样本决定了最终的最优超平面的位置；在式(5-104)和式(5-105)的加权求和中，实际也只有这些 $\alpha_i > 0$ 的样本参与求和。这些样本被称作支持向量(support vectors)，它们往往只是训练样本中的很少一部分。

对于这些支持向量来说，有

$$y_i [(\boldsymbol{w}^* \cdot \boldsymbol{x}_i) + b^*] - 1 = 0 \tag{5-107}$$

因为已经求出了 $\boldsymbol{w}^*$，所以 b^* 可以用任何一个支持向量根据式(5-107)求得。在实际的数值计算中，人们通常采用所有 α_i 非零的样本用式(5-107)求解 b^* 后再取平均。

最优超平面的思想是苏联学者 Vapnik 和 Chervonenkis 在 20 世纪 70 年代提出的[①]，20 世纪 90 年代由美国 AT&T 贝尔实验室 Vapnik 领导的小组对其进行了进一步发展，从 20 世纪 90 年代末开始在国际上迅速得到重视。由于最优超平面的解最后是完全由支持向量决定的，所以这种方法后来被称作支持向量机(support vector machines)，通常被简写为 SVM 或 SV 机。

这里介绍的只是线性可分情况下的线性支持向量机，更复杂的情况将在 5.8.3 节和第 6 章进一步介绍。

对比 5.5 节中的感知器算法，我们也可以把最优超平面等价地看作是在限制权值尺度的条件下求余量的最大化。感兴趣的读者可以自己尝试分析这一关系。

5.8.2 大间隔与推广能力

5.8.1 节从直观分析出发定义了最优超平面，即最大间隔分类超平面。那么，一个问题是，这样定义的超平面真的是最优吗？在什么意义下是最优的？在第 7 章中我们将介绍统计学习理论的核心思想和结论。由于一部分读者可能不需要掌握统计学习理论所涉及的理论内容，为了适应不同读者的需要，在本小节中，我们根据统计学习理论对这一问题进行简

① Vapnik V N, Chervonenkis A Ja. *Theory of Pattern Recognition*[俄文版]. Nauka, Mosco, 1974.

要说明。

从第1章的讨论已经知道，模式识别是一种基于数据的机器学习，学习的目的不仅是要对训练样本能够正确分类，而是要能够对所有可能的样本正确分类，这种能力叫做推广(generalization)。在线性可分情况下，我们用感知器算法(或其他算法)可以得到无数多种可能的解，它们都可以是训练误差为0，我们要追求的最优解应该是这些解中推广能力最大的解。

对于某个样本 $\boldsymbol{x}$，其真实的类别为 y，我们要用判别函数 $f(\boldsymbol{x},\boldsymbol{w})$ 来估计 y，定义这种估计带来的损失是 $L(y,f(\boldsymbol{x},\boldsymbol{w}))$，这里为了强调权值参数 $\boldsymbol{w}$ 对最后损失的影响，我们把它写为判别函数的自变量之一。那么，在某个 $\boldsymbol{w}$ 下对所有训练样本的分类决策损失就是

$$R_{\text{emp}}(\boldsymbol{w})=\frac{1}{N}\sum_{i=1}^{N}L(y_i,f(\boldsymbol{x}_i,\boldsymbol{w})) \tag{5-108}$$

称作经验风险。线性可分情况下，通过感知器算法，已经能使经验风险达到零。

但是，我们真正关心的是在权值 $\boldsymbol{w}$ 下未来所有可能出现的样本的错误率或风险，即

$$R(\boldsymbol{w})=\int L(y,f(\boldsymbol{x},\boldsymbol{w}))\mathrm{d}F(\boldsymbol{x},y) \tag{5-109}$$

称作期望风险。其中，$F(\boldsymbol{x},y)$ 表示所有可能出现的样本及其类别的联合概率模型。对比式(5-108)和式(5-109)可以知道，经验风险只是在给定的训练样本上对期望风险的估计。

那么，这样的估计准确吗？在多个使经验风险为0的解中，如何才能找到使期望风险最小的解？

由Vapnik等提出和发展的统计学习理论系统地回答了这一问题。

统计学习理论指出，有限样本下，经验风险与期望风险是有差别的，期望风险可能大于经验风险，但它们之间满足下面的规律

$$R(\boldsymbol{w})\leqslant R_{\text{emp}}(\boldsymbol{w})+\varphi\left(\frac{h}{N}\right) \tag{5-110}$$

其中，$\varphi(h/N)$ 称作置信范围，它与样本数 N 成反比，而与一个重要的参数 h 成正比。这个参数 h 是依赖于模式识别算法的设计的，称作VC维(VC Dimension)，它反映了所设计的学习机器(函数集)的复杂性，确切的定义请参考第7章和Vapnik所著的《统计学习理论的本质》或《统计学习理论》。

式(5-110)给出了有限样本下期望风险的上界。它告诉我们，在训练误差相同的情况下，学习机器的复杂度越低(VC维越低)，则期望风险与经验风险的差别就越小，因而学习机器的推广能力就越好。

在线性可分的问题中，我们能得到很多使 $R_{\text{emp}}(\boldsymbol{w})$ 为0的解，要使方法有最好的推广能力，就应该设法使 $\varphi(h/N)$ 最小。由于训练样本集是给定的，即 N 固定，能够调整的是算法的VC维。

统计学习理论中的另一个重要的结论是，对于规范化的分类超平面，如果权值满足 $\|\boldsymbol{w}\|\leqslant A$，那么这种分类超平面集合的VC维有下面的上界

$$h\leqslant\min([R^2A^2],d)+1 \tag{5-111}$$

其中，R^2 是样本特征空间中能包含所有训练样本的最小超球体的半径，d 是样本特征的维数。对于给定的样本集，这两项均是确定的。在求最大间隔分类超平面时，最大化分类间隔也就等价于最小化 A^2，实际上是使VC维上界最小。根据式(5-110)，这样就是试图使期望风险的置信范围尽可能小，即在经验风险都最小化为0的情况下追求期望风险的上界的最小化。

因此，支持向量机中最大分类间隔的准则，是为了通过控制算法的 VC 维实现最好的推广能力。在这个意义下，所得的分类超平面是最优的。

5.8.3 线性不可分情况

前面我们只讨论了线性可分情况下的最优超平面。线性不可分的情况下，上面定义的最优超平面不存在。本节我们来分析如何在这种情况下定义和求解支持向量机。

样本集不是线性可分，就是说对样本集

$$(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\cdots,(\boldsymbol{x}_N,y_N),\quad \boldsymbol{x}_i\in R^d,y_i\in\{+1,-1\} \tag{5-112}$$

不等式

$$y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]-1\geqslant 0,\quad i=1,2,\cdots,N \tag{5-113}$$

不可能被所有样本同时满足。

假定某个样本 $\boldsymbol{x}_k$ 不满足式(5-113)的条件，即 $y_k[(\boldsymbol{w}\cdot\boldsymbol{x}_k)+b]-1<0$，那么总可以在不等式的左侧加上一个正数 ξ_k，使得新的不等式 $y_k[(\boldsymbol{w}\cdot\boldsymbol{x}_k)+b]-1+\xi_k\geqslant 0$ 成立。

从这个思路出发，对每一个样本引入一个非负的松弛变量 ξ_i，$i=1,\cdots,N$，就可以把式(5-113)的不等式约束条件变为

$$y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]-1+\xi_i\geqslant 0,\quad i=1,2,\cdots,N \tag{5-114}$$

如果样本 $\boldsymbol{x}_j$ 被正确分类，即 $y_j[(\boldsymbol{w}\cdot\boldsymbol{x}_j)+b]-1\geqslant 0$，则 $\xi_j=0$；而如果有一个错分样本，则这个样本对应的 $y_j[(\boldsymbol{w}\cdot\boldsymbol{x}_j)+b]-1<0$，对应的松弛变量 $\xi_j>0$。

所有样本的松弛因子之和 $\sum_{i=1}^{N}\xi_i$ 可以反映在整个训练样本集上的错分程度，错分样本数越多，则 $\sum_{i=1}^{N}\xi_i$ 越大；同时，如果样本错误的程度越大(在错误的方向上远离分类面)，则 $\sum_{i=1}^{N}\xi_i$ 也越大。显然，我们希望 $\sum_{i=1}^{N}\xi_i$ 尽可能小。因此，可以在线性可分情况下的目标函数 $\frac{1}{2}\|\boldsymbol{w}\|^2$ 上增加对错误的惩罚项，定义下面的广义最优分类面的目标函数

$$\min_{\boldsymbol{w},b}\frac{1}{2}(\boldsymbol{w}\cdot\boldsymbol{w})+C\sum_{i=1}^{N}\xi_i \tag{5-115}$$

这个目标函数反映了我们的两个目标：一方面希望分类间隔尽可能大(对于分类正确的样本来说)，另一方面希望错分的样本尽可能少且错误程度尽可能低。参数 C 是一个常数，反映在这两个目标之间的折中。(注意，这里样本被错分的定义不是 $y_j[(\boldsymbol{w}\cdot\boldsymbol{x}_j)+b]<0$，而是 $y_j[(\boldsymbol{w}\cdot\boldsymbol{x}_j)+b]-1<0$，即第一类样本只要 $g(\boldsymbol{x})$ 小于 1 就算作错误，第二类样本只要 $g(\boldsymbol{x})$ 大于 -1 就算作错误。)

C 是一个需要人为选择的参数。通常，如果选择较小的 C，则表示对错误比较容忍而更强调对于正确分类的样本的分类间隔；相反，若选择较大的 C，则更强调对分类错误的惩罚。实际应用中，如果样本线性可分，则 C 的大小只是影响算法的中间过程而不影响最后结果，因为 $\sum_{i=1}^{N}\xi_i$ 最终会为 0。在线性不可分情况下，有时需要试用不同的 C 来达到更理想的结果。

下面把引入松弛因子后的广义最优分类面问题正式表述如下：在给定训练样本集

$$(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\cdots,(\boldsymbol{x}_N,y_N),\quad \boldsymbol{x}_i\in R^d,y_i\in\{+1,-1\} \tag{5-116}$$

的情况下，求解

$$\min_{\boldsymbol{w},b,\xi_i}\frac{1}{2}(\boldsymbol{w}\cdot\boldsymbol{w})+C\sum_{i=1}^{N}\xi_i \tag{5-117}$$

$$\text{s.t.}\qquad y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]-1+\xi_i\geqslant 0,i=1,2,\cdots,N \tag{5-118}$$

且

$$\xi_i\geqslant 0,\quad i=1,2,\cdots,N \tag{5-119}$$

与线性可分情况下的最优分类面类似，可以把这个问题转化为以下拉格朗日泛函的鞍点问题

$$\min_{\boldsymbol{w},b,\xi_i}\max_{\boldsymbol{\alpha}}L(\boldsymbol{w},b,\boldsymbol{\alpha})=\frac{1}{2}(\boldsymbol{w}\cdot\boldsymbol{w})+C\sum_{i=1}^{N}\xi_i-\sum_{i=1}^{N}\alpha_i\{y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]-1+\xi_i\}-\sum_{i=1}^{N}\gamma_i\xi_i \tag{5-120}$$

其中，$\alpha_i\geqslant 0,\gamma_i\geqslant 0$ 是对应式(5-118)和式(5-119)的拉格朗日乘子。

同样，把式(5-120)的拉格朗日泛函分别对 $\boldsymbol{w}$、b、ξ_i 求导并令其为0。经过一些简单的推导(读者可以作为课后练习)，可以得到广义最优分类面的对偶优化问题

$$\max_{\boldsymbol{\alpha}}Q(\boldsymbol{\alpha})=\sum_{i=1}^{N}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j y_i y_j(\boldsymbol{x}_i\cdot\boldsymbol{x}_j) \tag{5-121}$$

$$\text{s.t.}\qquad \sum_{i=1}^{N}y_i\alpha_i=0 \tag{5-122}$$

且

$$0\leqslant\alpha_i\leqslant C,\quad i=1,\cdots,N \tag{5-123}$$

原问题中的解向量满足

$$\boldsymbol{w}^*=\sum_{i=1}^{N}\alpha_i^*y_i\boldsymbol{x}_i \tag{5-124}$$

广义最优分类面的判别函数是

$$f(\boldsymbol{x})=\operatorname{sgn}\{g(\boldsymbol{x})\}=\operatorname{sgn}\{(\boldsymbol{w}^*\cdot\boldsymbol{x})+b\}=\operatorname{sgn}\left\{\sum_{i=1}^{N}\alpha_i^*y_i(\boldsymbol{x}_i\cdot\boldsymbol{x})+b^*\right\} \tag{5-125}$$

我们注意到，对偶问题式(5-121)～式(5-123)与线性可分情况下最优分类面的对偶问题式(5-101)～式(5-103)几乎相同，唯一不同的是在对 α_i 的约束条件式(5-123)中比式(5-103)多了一个上界 C。

根据库恩-塔克条件，式(5-98)的鞍点满足以下两套条件

$$\alpha_i\{y_i[(\boldsymbol{w}\cdot\boldsymbol{x}_i)+b]-1+\xi_i\}=0,\quad i=1,2,\cdots,N \tag{5-126}$$

$$\gamma_i\xi_i=(C-\alpha_i)\xi_i=0,\quad i=1,2,\cdots,N \tag{5-127}$$

从式(5-127)可以得到，只有对拉格朗日乘子达到上界 $\alpha_i=C$ 的样本才有 $\xi_i>0$，它们是被错分的样本(包括在两条平行的边界面之间的样本)，其余样本对应的 $\xi_i=0$。

而从式(5-126)得到，多数的 α_i 仍为0，只有

$$y_i[(\boldsymbol{w}\cdot\boldsymbol{x})+b]-1+\xi_i=0 \tag{5-128}$$

的样本才会使 $\alpha_i>0$。这些样本又分为两种情况，一种是分类正确但处在分类边界面上的样本，它们的 $0<\alpha_i<C,\xi_i=0$；另外一种则是分类错误的样本，它们的 $\alpha_i=C,\xi_i>0$。可以

用其中 $0<\alpha_i<C$ 的样本来通过式(5-128)求得 b。

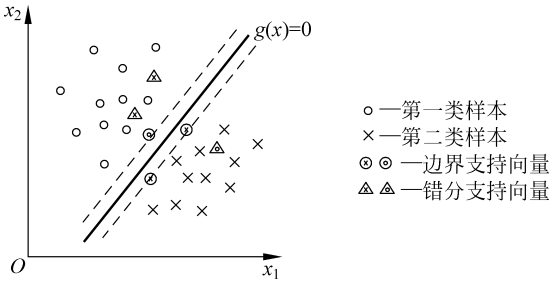


图 5-15 线性不可分情况下的广义最优分类面及其中的支持向量

需要说明的是,这两部分 $\alpha_i>0$ 的样本都是支持向量,但其含义与线性可分情况下已经不同。在某些文献里,把那些 $0<\alpha_i<C$ 的支持向量叫做边界向量(margin vectors)。图 5-15 中给出了两种不同的支持向量的例子。

由于广义最优分类面可以兼容线性可分情况下的最优分类面,所以人们通常采用的支持向量机都是考虑广义最优分类面的形式。

考查式(5-101)～式(5-103)和式(5-121)～式(5-123)的优化问题,可以发现目标函数式(5-101)、式(5-121)中只有 α_i 的二次项和一次项,这是一个对 α_i,$i=1,\cdots,N$ 在等式和不等式约束下的二次优化问题,具有唯一的极值点。关于具体的解法将在第 6 章介绍非线性的支持向量机后作简略介绍。

5.9 多类线性分类器

在前几节中讨论的都是两类的分类问题。在很多实际应用中,经常会面对多类的分类问题,例如在手写数字识别中,面对的是 0～9 十类。

解决多类分类问题有两种基本思路,一种方法是把多类问题分解成多个两类问题,通过多个两类分类器实现多类的分类;另一种方法是直接设计多类分类器。本节中我们讨论这两种多类分类方法中有代表性的线性方法。

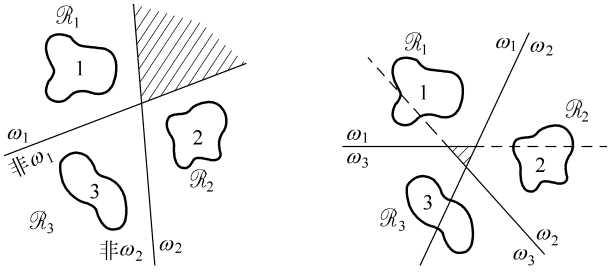
5.9.1 多个两类分类器的组合

假如要解决 0、1、2、3、4、5、6、7、8、9 这十个数字的识别问题,可以设计多个两类的分类器,例如,第一个分类器把“0”和其他数字分开,第二个分类器把“1”和其他数字分开……以此类推;或者,也可以这样设计多个两类分类器:用九个分类器分别把“0”和“1”、“0”和“2”、…、“0”和“9”分开,再用八个分类器分别把“1”和“2”、“1”和“3”、…、“1”和“9”分开……以此类推。这两种做法都可以最终实现把 0～9 十个数字分开,它们代表了用多个两类分类器构造多类分类器的两种典型的做法。

第一种做法叫做“一对多”的做法,英文可以叫 one-vs-rest 或者 one-over-all。假设共有 c 个类,$\omega_1,\omega_2,\cdots,\omega_c$,我们共需要 $c-1$ 个两类分类器就可以实现 c 个类的分类。

但是，这种做法可能会遇到两方面的问题。一个问题是，假如多类中各类的训练样本数目相当，那么，在构造每个一对多的两类分类器时会面临训练样本不均衡的问题，即两类训练样本的数目差别过大。虽然很多分类器算法并没有要求两类样本均衡，但是有些算法却可能会因为样本数目过于不均衡而导致分类面有偏，例如使得多数错误发生在样本数小的一类上。这在实际应用时需要注意，如果出现类似情况需要对算法采取适当的修正措施。

另一个问题是，用 $c-1$ 个线性分类器来实现 c 类分类，就是用 $c-1$ 个超平面来把样本所在的特征空间划分成 c 个区域，一般情况下，这种划分不会恰好得到 c 个区域，而是会多出一些区域，而在这些区域内的分类会出现歧义，如图 5-16(a)中的阴影部分所示。



(a) ω_2/非ω_1 (b) ω_1/ω_2

图 5-16 用多个两类分类器实现多类划分时可能出现的歧义区

第二种做法是对多类中的每两类构造一个分类器，称作“逐对”(pairwise)分类。考虑到把 ω_i 和 ω_j 分开与把 ω_j 和 ω_i 分开相同，对于 c 个类别，共需要$\frac{c(c-1)}{2}$个两类分类器。显然，这种做法要比一对多的做法多用很多两类分类器。但是，逐对分类不会出现两类样本数过于不均衡的问题，而且决策歧义的区域通常要比一对多分类器小，如图 5-16(b)中阴影部分所示。

在这里的讨论中，我们没有涉及具体的两类分类器是什么，只是假定每个分类器给出样本属于两类中任意一类的决策。实际上，很多分类器在最后的分类决策前得到的是一个连续的量，分类是对这个量用某个阈值划分的结果，例如所有线性分类器都是最后转化为一个线性判别函数 $g(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+w_0$ 与某一阈值(通常是 0)比较的问题。SVM 也是这样一种分类器。在很多线性分类器中，一个正确分类的样本，如果它离分类面越远，则往往对它的类别判断就更确定，因此可以把分类器的输出值看作对样本属于某一类别的一种打分，如果分值大于零(或其他阈值)则判断样本属于该类，而且分值越高对此分类越确信，反之决策不属于该类。

利用这种分类器，可以用 c 个一对多的两类分类器来构造多类分类系统，即每个类别对应一个分类器，其输出是对样本是否属于 ω_i 类给出一个判断。在多类决策时，如果只有一个两类分类器给出了大于阈值的输出，而其余分类器输出均小于阈值，则把这个样本分到该类。更进一步，如果各个分类器的输出是可比的，而且根据类别的定义知道任意样本必定属于且仅属于 c 个类别中的一类，那么可以在决策时直接比较各个分类器的输出，把样本赋予输出值最大的分类器所对应的类别。(但是需要注意，对很多分类器来说，如果它们是分别训练的，其输出值之间并不一定能保证可比性，在实际应用时需根据具体情况仔细

分析。)

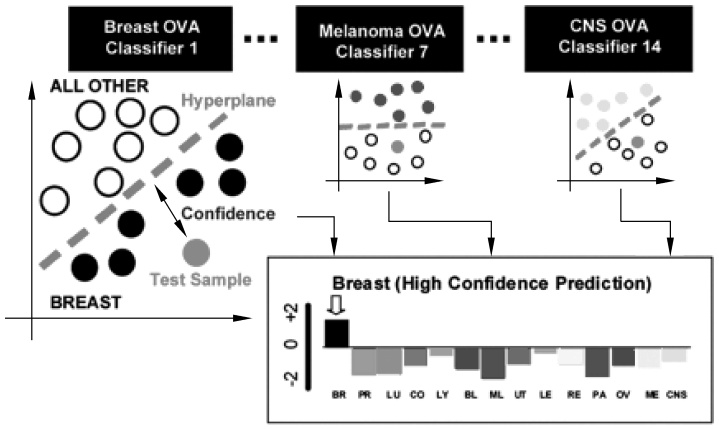


图 5-17 用多个两类 SVM 实现多类分类的例子[①]

图 5-17 给出了一个生物信息学中用多个 SVM 对基因芯片数据进行多种癌症的分类的例子[②]。在这个例子中,有 14 类癌症的基因表达数据,包括乳腺癌、肺癌、直肠癌、前列腺癌,等等。为了把这 14 类分开,他们对每一类癌症建立一个线性 SVM 分类器,把这类癌症与其他种类的癌症分开。这样共得到 14 个 SVM 分类器。在测试时,用这 14 个分类器分别对测试样本进行分类,哪个分类器给出最大的输出则把测试样本归到哪一类[②]。

除了以上两种划分方法,对于某些多类问题,如果人们对所研究的类别有较好的认识,能够根据类别间的内在关系把它们分级合并成多个两类分类问题,则可以用类似图 5-18 所示的二叉树来构建多个两类分类器。例如假如我们的目标是分出 a、b、c、d、e、f 六个类,如果发现这些类别的概念间有内在的关系,例如 e、f 两个类关系比较紧密,同属于一个更高层次的概念,c、d 同属于一个概念,b 和 c、d 又关系比较紧密,等等,则可以把问题分解成{a}对{b,c,d,e,f}、{b,c,d}对{e,f}、{b}对{c,d}、{c}对{d}、{e}对{f}这五个两类分类问题。

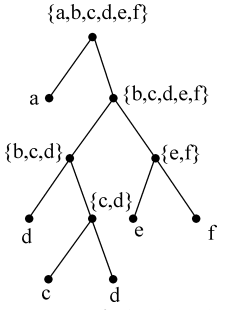


图 5-18 用二叉树把多类分类问题分解成多个两类问题

① Ramaswamy S, et al.. Multiclass cancer diagnosis using tumor gene expression signatures. *PNAS*, 2001, 98(26): 15149-15154.

② 注意,由于在 SVM 训练中要调整 $\boldsymbol{w}$ 的尺度以达到间隔最大,且保证离分类面最近的样本的输出值是 1,对不同的两类问题训练后的尺度并非完全相同。因此,这样得到的多个 SVM 的输出值之间严格来说并不能直接进行绝对值比较,只是 $g(\boldsymbol{x})$ 的符号有可比性。但是,在这个例子里,各个分类器间的尺度差别并不明显,所以直接使用多个 SVM 的输出进行比较就可以得到较好的效果。一般情况下,对于根据多个支持向量机的输出值来进行多类决策,还有一些理论问题需要进一步研究。

5.9.2 多类线性判别函数

所谓多类线性判别函数，是指对 c 类设计 c 个判别函数

$$g_i(\boldsymbol{x})=\boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{x}+w_{i0},\quad i=1,2,\cdots,c \tag{5-129}$$

在决策时哪一类的判别函数最大则决策为哪一类，即

$$\text{若 } g_i(\boldsymbol{x})>g_j(\boldsymbol{x}),\forall j\neq i,\quad \text{则 } \boldsymbol{x}\in\omega_i \tag{5-130}$$

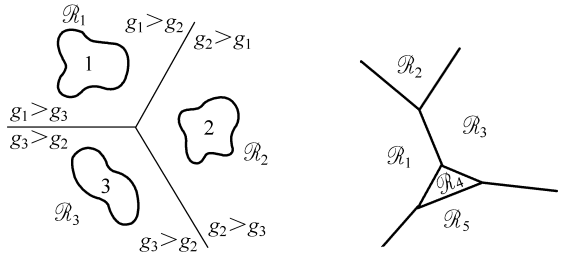
当然，我们也可以把这些判别函数表示成增广向量的形式

$$g_i(\boldsymbol{x})=\boldsymbol{\alpha}_i^{\mathrm{T}}\boldsymbol{y},\quad i=1,2,\cdots,c \tag{5-131}$$

其中，$\boldsymbol{\alpha}_i=\begin{bmatrix}\boldsymbol{w}_i\\ w_{i0}\end{bmatrix}$ 为增广权向量。

多类线性判别函数也称为多类线性机器，可以记作 $L(\boldsymbol{\alpha}_1,\boldsymbol{\alpha}_2,\cdots,\boldsymbol{\alpha}_c)$。

与上面讨论的用多个两类分类器进行多类划分的方法相比，多类线性机器可以保证不会出现有决策歧义的区域，如图 5-19 所示。



(a) 三类　　(b) 五类

图 5-19 多类线性机器

与两类情况下的感知器算法相同，这里首先考虑多类线性可分情况，即存在一个线性机器能够把所有样本都正确分类的情况。在这种情况下，可以用与感知器算法类似的单样本修正法来求解线性机器。具体算法如下：

(1) 任意选择初始的权向量 $\boldsymbol{\alpha}_i(0)$，$i=1,2,\cdots,c$，置 $t=0$。

(2) 考查某个样本 $\boldsymbol{y}^k\in\omega_i$，若 $\boldsymbol{\alpha}_i(t)^{\mathrm{T}}\boldsymbol{y}^k>\boldsymbol{\alpha}_j(t)^{\mathrm{T}}\boldsymbol{y}^k$，则所有权向量不变；若存在某个类 j，使 $\boldsymbol{\alpha}_i(t)^{\mathrm{T}}\boldsymbol{y}^k\leqslant\boldsymbol{\alpha}_j(t)^{\mathrm{T}}\boldsymbol{y}^k$，则选择 $\boldsymbol{\alpha}_j(t)^{\mathrm{T}}\boldsymbol{y}^k$ 最大的类别 j，对各类的权值进行如下的修正

$$\begin{cases}\boldsymbol{\alpha}_i(t+1)=\boldsymbol{\alpha}_i(t)+\rho_t\boldsymbol{y}^k\\ \boldsymbol{\alpha}_j(t+1)=\boldsymbol{\alpha}_j(t)-\rho_t\boldsymbol{y}^k\\ \boldsymbol{\alpha}_l(t+1)=\boldsymbol{\alpha}_l(t),\quad l\neq i,j\end{cases} \tag{5-132}$$

ρ_t 是步长，必要时可以随着 t 而改变。

(3) 如果所有样本都分类正确，则停止；否则考查另一个样本，重复(2)。

这一算法被称作逐步修正法(incremental correction)。可以证明，如果样本集线性可分，则该算法可以在有限步内收敛于一组解向量。

与感知器算法一样，当样本不是线性可分时，这种逐步修正法不能收敛，人们可以对算法作适当的调整而使算法能够停止在一个可以接受的解上，例如通过逐渐减小步长而强制使算法收敛。

同样，也可以像在感知器算法中那样引入余量，即把 $\boldsymbol{\alpha}_i(t)^{\mathrm{T}}\boldsymbol{y}^k > \boldsymbol{\alpha}_j(t)^{\mathrm{T}}\boldsymbol{y}^k$ 变为 $\boldsymbol{\alpha}_i(t)^{\mathrm{T}}\boldsymbol{y}^k > \boldsymbol{\alpha}_j(t)^{\mathrm{T}}\boldsymbol{y}^k + b$。

很多其他的两类分类算法都可以发展出相应的多类分类算法，但其中多数在实际中的应用并不广泛，所以在此不做更多介绍。在后面两章里还会看到更多的可以用于多类分类的算法。

5.9.3 多类罗杰斯特回归与软最大

在 5.7 节中讨论的罗杰斯特回归问题，所考虑的实际上是样本属于所关心的类和不属于所关心的类的问题。这个思路可以方便地推广到多类情况，对每一类考虑样本是否属于它。在这个视角下，式(5-133)的罗杰斯特函数

$$P(y=1 \mid x)=\frac{\mathrm{e}^{w_0+w_1 x}}{1+\mathrm{e}^{w_0+w_1 x}} \tag{5-133}$$

的分子可以看作是对样本属于该类的可能性的度量，而分母的作用则是把这个可能性归一化为概率。

把这个思路推广到多类情况，我们可以把模型设为样本属于每一类 j 都与一个参数为 w_j 的指数判别函数成正比，即

$$P(y=j \mid \boldsymbol{x}) \propto \mathrm{e}^{w_j \cdot \boldsymbol{x}}$$

用样本属于全部 c 个类别的判别函数做归一化，就得到

$$P(y=j \mid \boldsymbol{x})=\frac{\mathrm{e}^{w_j \cdot \boldsymbol{x}}}{\sum_{k=1}^{c}\mathrm{e}^{w_k \cdot \boldsymbol{x}}}, \quad j=1,\cdots,c \tag{5-134}$$

这个归一化指数函数在机器学习领域中被称作软最大(Softmax)函数，就是对样本的多类罗杰斯特回归。可以采用与两类罗杰斯特回归类似的思路用最大似然法求解。在第 12 章介绍深度学习时，我们还会看到软最大函数在多种深度神经网络中的应用。

5.10 讨论

线性判别函数是形式最简单的判别函数。它虽然算法简单，但是在一定条件下能够实现或逼近最优分类器的性能，因此在很多实际问题中得到了广泛的应用。而且，在很多情况下，虽然所研究的问题可能并不是线性的，但是由于我们所拥有的样本数目有限，或者样本观测中有较大噪声，我们可能仍然会使用线性分类器。这不但是一种在特定条件下追求“有限合理”解的妥协方案，更重要的是，在一些情况下，线性分类器可能比更复杂的模型取得更好的结果，尤其是更好的推广能力。

世界是非线性的,但很多情况下可以用线性来近似。

毕竟还有很多情况需要采用更复杂的非线性方法。在第6章我们将看到,很多非线性方法是以本章介绍的线性方法为基础发展起来的。例如,利用解决多类分类的思路可以设计多个分类器,用分段线性来逼近非线性;在本章中看到的最基本的感知器算法,就是一种最简单的人工神经元,人工神经网络中的多层感知器算法就是建立在它的基础上的;通过引入广义线性判别函数,可以把很多线性方法映射为非线性方法,而非线性的支持向量机则是通过用核函数的方法来实现广义线性判别函数。