4.7.2 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

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

在决策时哪一类的判别函数最大则决策为哪一类，即

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

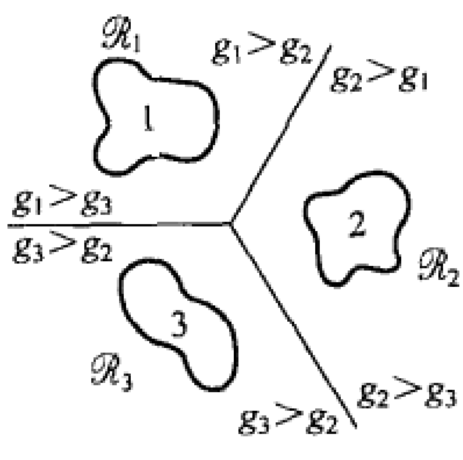
当然，我们也可以把这些判别函数表示成增广向量的形式

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

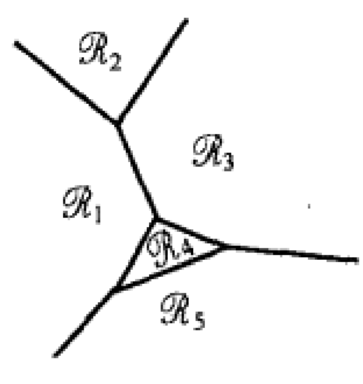
其中，$\boldsymbol{\alpha}_i = \begin{bmatrix} \boldsymbol{w}_i \\ w_{i0} \end{bmatrix}$为增广权向量。

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

与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



(a) 三类



(b) 五类

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

$$\begin{cases} \boldsymbol{\alpha}_i(t+1) = \boldsymbol{\alpha}_i(t) + \rho_t \mathbf{y}^k & \text{梯度上升} \\ \boldsymbol{\alpha}_j(t+1) = \boldsymbol{\alpha}_j(t) - \rho_t \mathbf{y}^k & \text{梯度下降} \\ \boldsymbol{\alpha}_l(t+1) = \boldsymbol{\alpha}_l(t), \quad l \neq i,j \end{cases}$$

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，比如通过逐渐减小步长而强制使算法收敛。