

# 一般的分段线性判别函数

有什么特点？

前面讲的是最小距离，现在不看距离。对每个子类定义一个判别函数，选判别函数值最大的类别。对每个子类建立更一般形式的线性判别函数，即，把每个类别划分成L个子类。

决策规则

$$\text{若 } g_k(x) = \max_{i=1, \dots, C} g_i(x)$$

则决策  $x \in W_k$

$$g_i(x) = \max_{l=1, \dots, l_i} g_i^l(x)$$

为什么  $g_i(x)$  分段？

因为判别函数在不同的位置，会采用不同的子分类器，取最大的，不看其他的，所以会分段

$$g_i^l(x) = w_{i1}^l \cdot x + w_{i0}^l$$

相邻两类的决策面

$$g_i(x) = g_j(x)$$

这个决策面也是由多个分段的超平面组成的，其中的一段是一类中的某个子类 and 另一类中的相邻子类之间的分类面。在确定了子类划分之后，分段线性判别函数的设计就等同于多类分类器的设计。

## 如何划分子类？

可以分为三种情况考虑。

第一种情况是,根据问题的领域知识和对数据分布的了解,人工确定子类的划分方案。

第二种情况,已知或者可以假定各类的子类数目,但是不知道子类的划分,可以用下面的错误修正法在设计分类器的同时确定出子类的划分。这里用增广的线性判别函数形式来描述这个算法。

第三种情况是子类数目无法事先确定。虽然可以用不同的子类数目尝试上面的算法,但是如果没有一个参考数字,盲目地试凑各种可能的子类数目所需要的运算量是巨大的。另一种方法是可以采用分类树的思想来分级划分子类和设计分段线性判别函数。

第二种情况：

(1) 初始化。任意给定各类各子类的权值 $\alpha_l^i(0)$ ,  $l=1,2,\dots,l_i$ ,  $i=1,2,\dots,c$ ,通常可以选用小的随机数。

(2) 在时刻  $t$ ,当前权值为 $\alpha_l^i(t)$ ,  $l=1,2,\dots,l_i$ ,  $i=1,2,\dots,c$ ,考虑某个训练样本  $y_k \in \omega_j$ ,找出  $\omega_j$  类的各子类中判别函数最大的子类,记为  $m$ ,即

$$\alpha_j^m(t)^T y_k = \max_{l=1,\dots,l_j} \{\alpha_j^l(t)^T y_k\} \quad (5-7)$$

考查当前权值对样本  $y_k$  的分类情况：

① 若 $\alpha_j^m(t)^T y_k > \alpha_l^i(t)^T y_k$ ,  $\forall i=1,\dots,c, i \neq j, l=1,\dots,l_i$ ,即  $y_k$  分类正确,则所有 $\alpha_l^i(t)$ 均不变: $\alpha_l^i(t+1) = \alpha_l^i(t)$ ,  $l=1,2,\dots,l_i, i=1,2,\dots,c$ ;

② 若对某个  $i \neq j$ ,存在子类  $l$  使得 $\alpha_j^m(t)^T y_k \leq \alpha_l^i(t)^T y_k$ ,即  $y_k$  被当前权值错分,则选取 $\alpha_l^i(t)^T y_k$ 中最大的子类(不妨记作  $\omega_i$  类的第  $n$  个子类),对权值进行如下修正

$$\alpha_j^m(t+1) = \alpha_j^m(t) + \rho_t y_k \quad (5-8a)$$

$$\alpha_l^n(t+1) = \alpha_l^n(t) - \rho_t y_k \quad (5-8b)$$

其余权值不变。

(3)  $t=t+1$ ,考查下一个样本,回到第(2)步。如此迭代,直到算法收敛。

(1) 初始化

(2) 从第 $j$ 类取一个样本, 求每个子类的判别函数值, 选最大值

因为每个类里都有多个子类, 用当前的最大值与所有的其他类的子类的判别函数比较, 如果还是最大, 则不必更新参数, 如果不是最大, 则分别对当前子类进行更新。

(3) 考察所有样本

算法的终止条件是算法收敛, 即对所有训练样本都分类正确, 在一轮循环中不再对权值进行修正。从上一章关于感知器和多类线性判别函数的逐步修正法的讨论可以知道, 这种算法只有在不同类别的各个子类之间都是线性可分的情况下才能保证收敛, 对某些数据并不一定能实现, 在指定子类数目时更是如此。与第4章中讨论的方法类似, 如果算法不能收敛, 人们通常可以用逐步缩小训练步长 $\rho$ 的方法强制算法收敛。

可以看出, 这个算法与4.7.2节介绍的多类线性判别函数的逐步修正法很相像, 这里的子类相当于4.7.2节中考虑的多类中的一类。所不同的是, 这里的分类器设计过程实际上也是子类的划分过程, 而考查权值是否需要修正时并不是考查样本是否被分到某个特定的子类, 而是只需要判断样本是否被分到它所属的类别的几个子类中的一个。

模型更新子分类器的参数就相当于改变了子类的划分, 只是子类数量没改变。

第三种情况:

我们可先用两类线性判别函数算法找一个权向量 $a$ , 它所对应的超平面把整个样本集分成两部分, 我们称之为样本子集。由于样本集不是线性可分的, 因而每一部分仍然包含两类样本。接着, 再利用算法找出第二个权向量、第三个权向量, 超平面分别把相应的样本子集分成两部分。若某一部分仍然包含两类样本, 则继续上述过程, 直到某一权向量把两类样本完全分开为止。这样得到的分类器显然也是分段线性的。

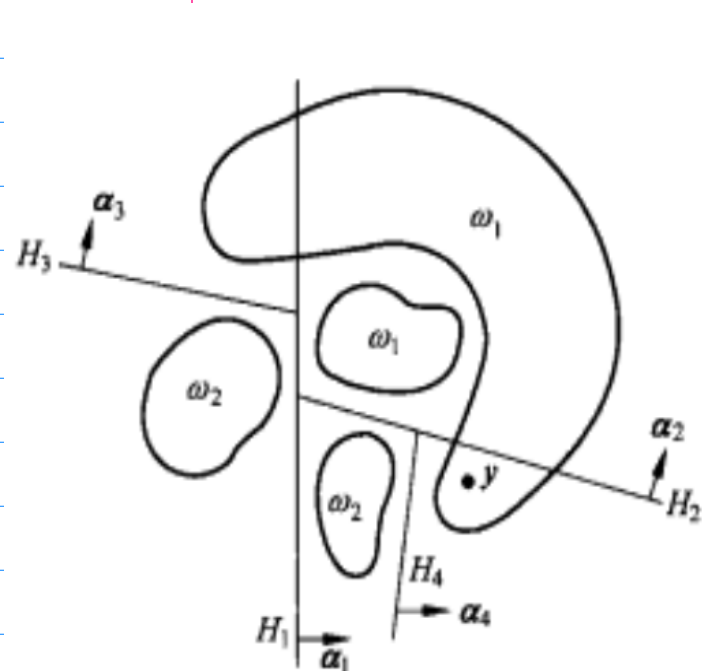


图 5-4 树状分段性分类器举例

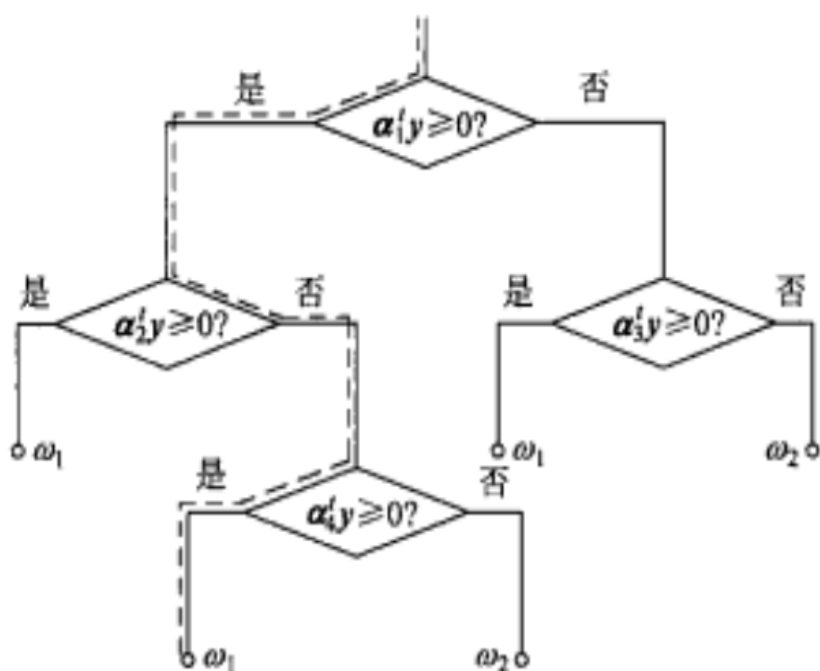


图 5-5 与图 5-4 对应的树状决策过程

需要指出,这种方法对初始权向量的选择很敏感,其结果随初始权向量的不同而大不相同。此外,在每个节点上所用的寻找权向量的方法不同,结果也将各异。通常可以选择分属两类的欧氏距离最小的一对样本,取其垂直平分面的法向量作为初始值,然后求得局部最优解作为第1段超平面的法向量。对包含两类样本的各子类的划分也可以采用同样的方法。

淘宝店铺-酷流科技 掌柜：我是雷锋的朋友