目录

第1章　绪 论

第2章 数学基础

第3章 贝叶斯决策

第4章 线性模型

第5章 支持向量机

尽管逻辑回归模型为线性分类问题提供了一整套完善的优化求解模型，但从原理上看，逻辑回归的最优性是以单类样本服从正态分布为前提的（详见**错误!未找到引用源。**），而在实际应用中，这一前提通常难以得到保证。我们是否能够从传统的判别模型的思想上再向前一步，摆脱后验概率模型对分类器的束缚，建立一种更加直观和有效的判别方法呢？如果我们遇到的是非线性分类问题，又当如何呢？支持向量机给了上述问题一个非常有效的解答思路，这正是本章需要介绍和讨论的内容。

5.1 线性可分 SVM 问题原型

SVM 的全称是 **Support Vector Machine**，即**支持向量机**，主要用于解决模式识别领域中的数据分类问题，属于有监督学习算法的一种。SVM 要解决的问题可以用一个经典的二分类问题加以描述。如图 5-1 所示，红色和蓝色的二维数据点显然是可以被一条直线分开的，在模式识别领域称为**线性可分问题(Linear Separable Problem)**。然而将两类数据点分开的直线显然不止一条。图 5-1(b)和(c)分别给出了 A、B 两种不同的分类方案，其中黑色实线表示二维空间中的“决策面”。每个决策面对应了一个线性分类器。虽然从图中所示的数据上看，这两个分类器的分类结果是一样的，但如果考虑到还未观测的潜在数据，两者的分类性能显然是有差别的。

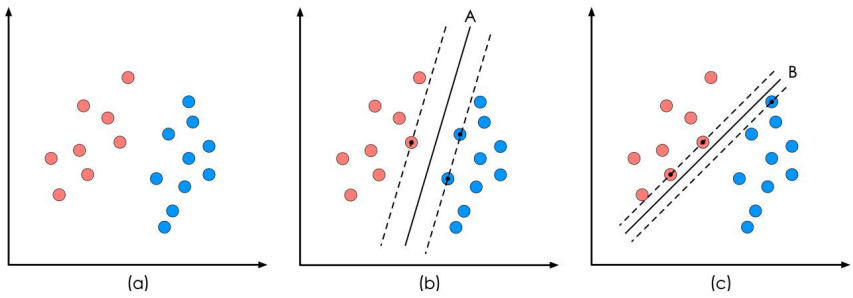

图 5-1 二分类问题描述

SVM 算法认为图 5-1 中的分类器 A 在性能上优于分类器 B，其依据是 A 的分类间隔比 B 要大。这里涉及到第一个 SVM 独有的概念**——分类间隔(Margin)**。在保证决策面方向不变且不会出现错分样本的情况下移动决策面，会在原来的决策面两侧找到两个极限位置（越过该位置就会产生错分现象），如虚线所示。虚线的位置由决策面的方向和距离原决策面最近的几个样本的位置决定。而这两条平行虚线正中间的分界线就是在保持当前决策面方向不变的前提下的最优决策面。两条虚线之间的垂直距离就是这个最优决策面对应的分类间隔。显然每一个可能把数据集正确分开的方向都有一个最优决策面（有些方向无论如何移动决策面的位置也不可能将两类样本完全分开），而不同方向的最优决策面的分类间隔通常是不同的，

那个具有**最大间隔(Maximum Margin)**的决策面就是 SVM 要寻找的最优解。而这个真正的最优解对应的两侧虚线所穿过的样本点，就是 SVM 中的支持样本点，称为“支持向量”。对于图 5-1 中的数据，A 决策面就是 SVM 寻找的最优解，而相应的三个位于虚线上的样本点在坐标系中对应的向量就叫做**支持向量(Support Vector)**。

从表面上看，我们优化的对象似乎是这个决策面的方向和位置。但实际上最优决策面的方向和位置完全取决于选择哪些样本作为支持向量。而在经过漫长的公式推导后，最终会发现其实与线性决策面的方向和位置直接相关的参数都会被约减掉，最终结果只取决于支持向量的选择结果。

到这里，我们明确了 SVM 算法要解决的是一个最优分类器的设计问题。既然叫作最优分类器，其本质必然是个最优化问题。所以，接下来我们要讨论的就是如何把 SVM 变成用数学语言描述的最优化问题模型，这里将会利用到**错误!未找到引用源。**节最优化方法中介绍的有约束优化方法，包括拉格朗日乘子法、KKT 条件和拉格朗日对偶等概念和方法，因此读者在继续阅读之前应先结合 5.2 节 SVM 的模型，重温上述最优化方法的相关内容。

5.2 线性可分 SVM 问题的数学模型

SVM 问题从数学原型上看属于有约束最优化问题，该问题通常包含三个最基本的要素：1）目标函数，也就是你希望什么指标达到最好；2）优化对象，你期望通过改变哪些因素来使你的目标函数达到最优；3）约束条件，最优解必须满足的条件，通常与任务要求有关。在线性 SVM 算法中，目标函数显然是“分类间隔”，而优化对象则是决策面方程的参数，至于约束条件，则要求 SVM 模型在线性可分问题中能够正确分类所有的训练样本。所以要对 SVM 进行数学建模，首先要对上述三个要素（“分类间隔”和“决策面”）进行数学描述。按照一般的思维习惯，我们先描述决策面方程。

5.2.1 决策面方程

为了建立更加直观的印象，首先考虑在二维空间中用一根直线将两类二维样本分开的情况，先从我们熟悉的二维直角坐标系下的直线方程开始，描述决策线（决策面的二维表示）。

$$y = ax + b \tag{5.1}$$

将x轴记为x_1轴，y轴记为x_2轴，于是公式(5.1)中的直线方程可以改写为：

$$x_2 = ax_1 + b \tag{5.2}$$

公式(5.2)可以表示为向量形式：

$$[a, -1]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + b = 0 \tag{5.3}$$

考虑到我们在等式两边乘上任何实数都不会改变等式的成立，所以我们可以写出一个更加一般的向量表达形式:

$$g(\boldsymbol{x}) = \boldsymbol{\omega}^T\boldsymbol{x} + \gamma = 0 \tag{5.4}$$

其中，变量$\boldsymbol{\omega}, \boldsymbol{x}$是向量，$\boldsymbol{\omega} = [\omega_1, \omega_2]^T, \boldsymbol{x} = [x_1, x_2]^T$。公式(5.4)显然是一个线性分类器的决策面方程，$g(\boldsymbol{x})$为判别函数。结合**错误!未找到引用源。**节的内容，可知向量$\boldsymbol{\omega}$垂直于决策面，换言之，决策面在空间中的方向是由向量$\boldsymbol{\omega}$控制的。与之相对应的，γ是截距，它控制了决策面的位置。显然，寻找一个最优决策面就是参数$\boldsymbol{\omega}$和γ的优化过程。结合前面的分析，一旦决策面的方向确定，则最优决策面的位置也就确定了，因此这两组参数中比较重要的是参数$\boldsymbol{\omega}$的优化。

在讨论完二维的情况，我们可以考虑 n 维的情况。显然，在n维空间中的$n-1$维的决策面方程也遵循公式(5.4)的形式，区别在于向量$\boldsymbol{\omega}, \boldsymbol{x}$的维度从原来的 2 维变成了$n$维。

5.2.2 分类“间隔”的计算

根据 SVM 算法的设计思想，分类间隔是整个优化问题的目标函数，因此首先需要给出分类间隔的函数形式。从几何形态上看，分类间隔是支持向量样本到决策面的距离的二倍，如图 5-2 所示。

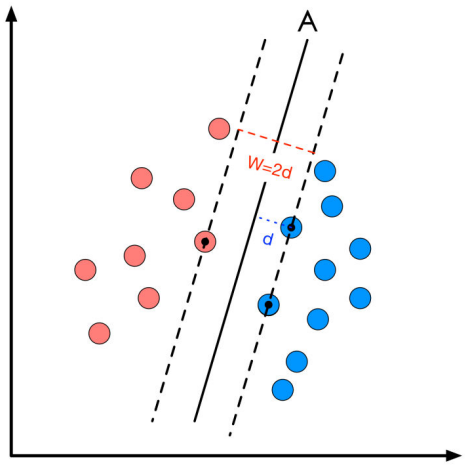

图 5-2 分类间隔计算

因此分类间隔W可以利用点$\boldsymbol{x}$到决策面的距离公式进行计算，如公式(5.5)所示：

$$W = 2d = \frac{2|\boldsymbol{\omega}^T\boldsymbol{x} + \gamma|}{\|\boldsymbol{\omega}\|} \tag{5.5}$$

其中$\|\boldsymbol{\omega}\|$是向量$\boldsymbol{\omega}$的模，表示在空间中向量的长度。$\boldsymbol{x} = [x_1, x_2, \ldots, x_d]^T$表示支持向量样本点的坐标。$\boldsymbol{\omega}, \gamma$是决策面方程的参数。因此，只要能够找到一组满足约束条件的参数$\boldsymbol{\omega}, \gamma$使得分类间隔$W$最大化，即相当于解决了 SVM 问题。

5.2.3 约束条件

接着 5.2.2 节的结尾，显然并不是所有的$\boldsymbol{\omega}, \gamma$都能够满足当前的分类任务要求的，所以需

要讨论一下约束条件的问题：

1）并不是所有的方向都存在能够实现 100%正确分类的决策面，我们如何判断某个方向上是否存在能够将所有的样本点都正确分类的决策面呢？

2）即便找到了正确的决策面方向，还要注意决策面的位置应该在间隔区域内，所以用来确定决策面位置的截距γ也不能自由地优化，而是受到决策面方向和样本点分布的约束。

3）即便取到了合适的方向和截距，公式(5.5)里面的$\boldsymbol{x}$也不是随随便便的一个样本点，而是支持向量对应的样本点。对于一个给定的决策面，我们该如何找到对应的支持向量呢？

尽管上面看起来是 3 个约束条件，但 SVM 算法通过一些巧妙的数学技巧，可以将三个约束条件融合在一个不等式内。

我们首先考虑一个决策面是否能够将所有的样本都正确分类的约束。图 5-2 中的样本点分成两类（红色和蓝色），我们为每个样本点$\boldsymbol{x}_i$加上一个类别标签y_i：

$$y_i = \begin{cases} +1 & \text{for 蓝色样本} \\ -1 & \text{for 红色样本} \end{cases} \tag{5.6}$$

如果一个决策面方程能够完全正确地对图 5-2 中的样本点进行分类，就会满足下面的公式：

$$\begin{cases} \boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma > 0 & \forall y_i = 1 \\ \boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma < 0 & \forall y_i = -1 \end{cases} \tag{5.7}$$

如果我们要求再高一点，假设决策面正好处于间隔区域的中轴线上，并且相应的支持向量对应的样本点到决策面的距离为 d，那么公式(5.7)就可以进一步写成：

$$\begin{cases} \dfrac{\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma}{\|\boldsymbol{\omega}\|} \geq d, & \forall y_i = 1 \\ \dfrac{\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma}{\|\boldsymbol{\omega}\|} \leq -d, & \forall y_i = -1 \end{cases} \tag{5.8}$$

令两个不等式的左右两边都除 d，可得到：

$$\begin{cases} \boldsymbol{\omega}_d^T \boldsymbol{x}_i + \gamma_d \geq 1, & \forall y_i = 1 \\ \boldsymbol{\omega}_d^T \boldsymbol{x}_i + \gamma_d \leq -1, & \forall y_i = -1 \end{cases} \tag{5.9}$$

其中

$$\boldsymbol{\omega}_d = \frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\| d}, \qquad \gamma_d = \frac{\gamma}{\|\boldsymbol{\omega}\| d} \tag{5.10}$$

把$\boldsymbol{\omega}_d$和γ_d就当成一条直线的方向矢量和截距。你会发现决策面的方向和位置没有发生任何变化，直线$\boldsymbol{\omega}_d^T \boldsymbol{x}_i + \gamma_d = 0$和直线$\boldsymbol{\omega}^T \boldsymbol{x} + \gamma = 0$表示同一个超平面。现在，忘记原来的直线方程参数$\boldsymbol{\omega}$和$\gamma$，给参数$\boldsymbol{\omega}_d$和$\gamma_d$重新命名为$\boldsymbol{\omega}$和$\gamma$。我们可以直接说：“对于线性可分的两类样本点，可以找到某些决策面满足下面的条件：

$$\begin{cases} \boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma \geq 1, & \forall y_i = 1 \\ \boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma \leq -1, & \forall y_i = -1 \end{cases} \tag{5.11}$$

公式(5.11)是 SVM 优化问题的约束条件的基本描述。

5.2.4 线性 SVM 优化问题的数学模型

观察公式(5.11)，结合 SVM 问题的定义，不难理解只有当$\boldsymbol{x}_i$是决策面$\boldsymbol{\omega}^T \boldsymbol{x} + \gamma = 0$所对应的支持向量时，公式(5.11)中的等号才会成立。这一点启发我们去进一步简化目标函数。回头看看公式(5.5)，你会发现等式右边分子部分的绝对值符号内部的表达式$\boldsymbol{\omega}^T \boldsymbol{x} + \gamma$正好跟公式(5.11)中两个不等式左边的表达式完全一致，无论原来这些表达式的值是 1 或者-1，其绝对值都是 1。所以对于这些支持向量，有：

$$d = \frac{|\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma|}{\|\boldsymbol{\omega}\|} = \frac{1}{\|\boldsymbol{\omega}\|} \tag{5.12}$$

上式的几何意义是，支持向量样本点到决策面方程的距离就是$1/\|\boldsymbol{\omega}\|$。我们原来的任务是找到一组参数$\boldsymbol{\omega}$,$\gamma$使得分类间隔$W = 2d$最大化，根据公式(5.12)就可以转变为$\|\boldsymbol{\omega}\|$的最小化问题，也等效于$\|\boldsymbol{\omega}\|^2/2$的最小化问题。

另外我们还可以尝试将公式(5.12)给出的约束条件进一步在精练，把类别标签y_i和两个不等式左边相乘，形成统一的表述：

$$y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma) \geq 1, \qquad \forall \boldsymbol{x}_i \tag{5.13}$$

至此可以给出线性 SVM 最优化问题的数学描述如下：

$$\begin{aligned} &\min_{\boldsymbol{\omega},\gamma} \frac{1}{2}\|\omega\|^2 \\ &\text{s.t. } y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma) \geq 1, \qquad i = 1,2,\dots,m \end{aligned} \tag{5.14}$$

其中，m 是样本总数。公式(5.14)描述的是一个典型的不等式约束条件下的二次型函数优化问题，同时也是支持向量机的基本数学模型。需要注意的是，在公式(5.14)中，所有的$\boldsymbol{x}_i, y_i$作为训练集中的样本和标签都是已知的，因此对于每一个样本，存在一个线性不等式约束条件。而目标函数$\|\omega\|^2/2$是一个典型的凸函数，所以 SVM 问题是多个线性不等式约束条件下的凸优化问题，参考**错误!未找到引用源。**节的内容，发现这类问题可以使用拉格朗日对偶的方法进行求解。

5.3 线性可分 SVM 问题求解

5.3.1 线性可分 SVM 问题的转化

观察公式(5.14)，作为一个具有多个不等式约束条件的凸优化问题，SVM 问题的拉格朗日函数可以写为：

$$L(\omega,\gamma,\boldsymbol{\alpha})=\frac{\|\boldsymbol{\omega}\|^2}{2}+\sum_{i=1}^{m}\alpha_i(1-y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i+\gamma)) \tag{5.15}$$

其中，$\boldsymbol{\omega}=[\omega_1,\omega_2,\dots,\omega_d]^T,\boldsymbol{\alpha}=[\alpha_1,\alpha_2,\dots,\alpha_m]^T$。该拉格朗日函数最优化的原始问题为：

$$\min_{\boldsymbol{\omega},\gamma}\left[\max_{\boldsymbol{\alpha}:\alpha_i\geq 0}L(\boldsymbol{\omega},\gamma,\boldsymbol{\alpha})\right] \tag{5.16}$$

相应的拉格朗日对偶问题为：

$$\max_{\boldsymbol{\alpha}:\alpha_i\geq 0}\left[\min_{\boldsymbol{\omega},\gamma}L(\boldsymbol{\omega},\gamma,\boldsymbol{\alpha})\right] \tag{5.17}$$

根据**错误!未找到引用源。**节的推导，对拉格朗日对偶问题进行求解，首先求解：

$$\min_{\boldsymbol{\omega},\gamma}L(\boldsymbol{\omega},\gamma,\boldsymbol{\alpha})=\min_{\boldsymbol{\omega},\gamma}\left[\frac{1}{2}\|\boldsymbol{\omega}\|^2+\sum_{i=1}^{m}\alpha_i(1-y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i+\gamma))\right] \tag{5.18}$$

为了求拉格朗日函数的极小值，分别令函数$L(\omega,\gamma,\boldsymbol{\alpha})$ 对$\boldsymbol{\omega},\gamma$ 求偏导，并使其等于 0。

$$\frac{\partial L}{\partial\boldsymbol{\omega}}=\mathbf{0}\Longrightarrow\boldsymbol{\omega}=\sum_{i=1}^{m}\alpha_i y_i\boldsymbol{x}_i \tag{5.19}$$

$$\frac{\partial L}{\partial\gamma}=0\Longrightarrow 0=\sum_{i=1}^{m}\alpha_i y_i \tag{5.20}$$

将公式(5.19)和(5.20)带入公式(5.18)，可以得到：

$$\min_{\boldsymbol{\omega},\gamma}L(\boldsymbol{\omega},\gamma,\boldsymbol{\alpha})=\frac{1}{2}\left(\sum_{i=1}^{m}\alpha_i y_i\boldsymbol{x}_i\right)^T\left(\sum_{i=1}^{m}\alpha_i y_i\boldsymbol{x}_i\right)+\sum_{i=1}^{m}\alpha_i-\sum_{i}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\boldsymbol{x}_i^T\boldsymbol{x}_j \tag{5.21}$$

根据多项式乘法的基本规律——所有项和的积等于所有项积的和，有：

$$\left(\sum_{i=1}^{m}\alpha_i y_i\boldsymbol{x}_i\right)^T\left(\sum_{i=1}^{m}\alpha_i y_i\boldsymbol{x}_i\right)=\sum_{i}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\boldsymbol{x}_i^T\boldsymbol{x}_j \tag{5.22}$$

则公式(5.21)可以化简为：

$$\min_{\boldsymbol{\omega},\gamma}L(\boldsymbol{\omega},\gamma,\boldsymbol{\alpha})=\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j\boldsymbol{x}_i^T\boldsymbol{x}_j \tag{5.23}$$

将公式(5.23)带入公式(5.17)，对目标函数取反，最大化问题变为最小化问题，则线性 SVM 的拉格朗日对偶问题，可以写为：

$$\left[\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\qquad\qquad\sum_{i=1}^{m}\right] \tag{5.24}$$

$$s.t.\sum_{i=1}^{m}\alpha_i y_i = 0,\ \alpha_i \geq 0, i = 1,2,\dots,m$$

显然，只要能够求解公式(5.24)描述的最优化问题，就能够实现 SVM 问题的求解。

然而相比于公式(5.14)描述的 SVM 优化问题，公式(5.24)的目标函数和约束条件在形式上更加复杂，解决这一问题还需要综合利用最优化方法中的 KKT 条件和一种称为 SMO 的专用优化算法。

5.3.2 SVM 求解的 KKT 条件

让我们结合 SVM 的实际情况，回顾**错误!未找到引用源。**节的**错误!未找到引用源。**，则公式(5.14)中的线性不等式约束条件对应的最优解的 KKT 条件可以写为：

$$\begin{cases}\alpha_i \geq 0 \\ y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 \geq 0 \\ \alpha_i(y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1) = 0\end{cases}, i = 1,2,\dots,m \tag{5.25}$$

上式蕴含的意思是，对于任意训练样本$\boldsymbol{x}_i$，或者$\alpha_i = 0$，或者$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) = 1$，两者总有一个成立。前面分析过，$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) = 1$意味着样本$\boldsymbol{x}_i$是支撑向量。所以可以得到这样一个结论，那就是：只有支撑向量$\boldsymbol{x}_i$对应的拉格朗日乘子$\alpha_i > 0$。观察公式(5.24)中的目标函数，可以看出最终的目标函数仅由$\alpha_i > 0$的样本决定，即仅由支撑向量决定。

之所以要讨论 SVM 问题的 KKT 条件，是因为 KKT 条件是最优解的充分必要条件。换言之，如果能找到一组$\alpha_i, i = 1,2,\dots,m$完全满足 KKT 条件，它们就是 SVM 问题的最优解。所以求解 SVM 优化问题在思路上可以转换为寻找一组满足 KKT 条件的解。

5.3.3 SMO 算法

SMO 算法是 **Sequential Minimal Optimization** 的缩写，即**序列最小优化**算法，是一种启发式优化算法。该算法以满足 KKT 条件为目标，将多个变量的综合优化问题，转化为一系列单个变量优化的迭代过程。具体的算法过程，分为以下 4 个步骤：

1) 变量选择：选择对加速优化进程最有利的一对拉格朗日乘子；
2) 问题转化：将 SVM 的多变量优化问题转化为二元优化问题；
3) 优化求解：找到满足约束条件的二元优化问题的最优解
4) 迭代与终止：重复步骤 1)~3)，直到所有的变量均满足 KKT 条件停止。

然而在算法设计过程中，变量选择的目的是为了使优化进程更快，因此需要比较单次优化前后的变化来选择最合适的一对拉格朗日乘子，因此需要先推导出单次变量优化的结果，才能给出变量选择的规则。鉴于此，本文先介绍问题转化与单变量寻优，之后介绍变量选择策略与终止条件。

(1) 问题转化

首先初始化m个拉格朗日乘子，令$\alpha_i(t = 0) = 0, i = 1,2,\dots,m$。之后，在每次迭代优化之前选择 2 个乘子$\alpha_j$和$\alpha_k$作为当前待优化的变量，锁定其他没有选中的拉格朗日乘子$\alpha_i, \forall i \neq j,k$。则公式(5.24)描述的$m$个变量的最优化问题，将会初步转化为$\alpha_j$和$\alpha_k$的二元优化问题。考虑到$y_i = +1$或-1，所以有$y_i^2 = 1, \forall i = 1,2,\dots,m$，则优化问题的目标函数和约束条件等价于

公式(5.26)所示：

$$\min_{\alpha_j,\alpha_k}\left[\frac{1}{2}\left(\alpha_j^2K_{jj}+\alpha_k^2K_{kk}\right)+\alpha_j\alpha_ky_jy_kK_{jk}+y_j\alpha_jv_j+y_k\alpha_kv_k-\left(\alpha_j+\alpha_k\right)\right]$$
$$\text{s.t. } y_j\alpha_j+y_k\alpha_k=-\sum_{i\neq j,k}y_i\alpha_i=C,\ \alpha_j,\alpha_k\geq 0 \tag{5.26}$$

其中

$$K_{jk}=K\left(\boldsymbol{x}_j,\boldsymbol{x}_k\right)=\boldsymbol{x}_j^T\boldsymbol{x}_k,\qquad j,k=1,2,\dots,m \tag{5.27}$$

$$v_j=\boldsymbol{x}_j^T\sum_{i\neq j,k}\alpha_iy_i\boldsymbol{x}_i=\sum_{i\neq j,k}\alpha_iy_iK_{ij},\ v_k=\boldsymbol{x}_k^T\sum_{i\neq j,k}\alpha_iy_i\boldsymbol{x}_i=\sum_{i\neq j,k}\alpha_iy_iK_{ik} \tag{5.28}$$

(2) 优化求解

观察公式(5.26)的线性约束条件，同时考虑$y_j^2=y_k^2=1$，推出α_j可以写成α_k的表达式：

$$\alpha_j=y_j(C-y_k\alpha_k) \tag{5.29}$$

则两个变量α_j和α_k转化为单一变量α_k。进一步考虑约束条件$\alpha_j\geq 0$，则有

$$\begin{aligned}y_j(C-y_k\alpha_k)&\geq 0\\ y_jy_k\alpha_k&\leq y_jC\end{aligned} \tag{5.30}$$

进而推出两种情况：

$$\alpha_k\begin{cases}\leq y_jC, & \text{if } y_ky_j=1\\ \geq y_jC, & \text{if } y_ky_j=-1\end{cases} \tag{5.31}$$

再结合 $\alpha_k\geq 0$的要求，可以进一步限定α_k的取值范围。

情况 1：当$y_ky_j=1$时，$\alpha_k\in[0,y_jC]$，如果此时$y_kC<0$，则α_k的可行解区域为空,则需要重新选取α_k；

情况 2：当$y_ky_j=-1$时，$\alpha_k\in\left[\max\left(0,y_jC\right),+\infty\right)$；

在明确了上述取值范围后，将公式(5.29)带入公式(5.26)，则目标函数变为一个α_k的单变量函数$L(\alpha_k)$：

$$\begin{aligned}L(\alpha_k)=\frac{1}{2}&\left((C-y_k\alpha_k)^2K_{jj}+\alpha_k^2K_{kk}\right)+y_k\alpha_k(C-y_k\alpha_k)K_{jk}+(C-y_k\alpha_k)v_j+y_k\alpha_kv_k\\ &-\alpha_k-y_k(C-y_k\alpha_k)\end{aligned} \tag{5.32}$$

对$L(\alpha_j)$求导，得到：

$$\frac{\partial L}{\partial\alpha_k}=\alpha_kK_{jj}+\alpha_kK_{kk}-2\alpha_kK_{jk}-Cy_kK_{jj}+Cy_kK_{jk}+y_kv_k-y_kv_j-1+y_ky_j \tag{5.33}$$

令该导数为 0，求解α_k^*：

$$\alpha_k^*\left(K_{jj}+K_{kk}-2K_{jk}\right)=y_k\left(C\left(K_{jj}-K_{jk}\right)+\left(v_j-v_k\right)+\left(y_k-y_j\right)\right)$$
$$\alpha_k^*=\frac{y_k\left(C\left(K_{jj}-K_{jk}\right)+\left(v_j-v_k\right)+\left(y_k-y_j\right)\right)}{\left(K_{jj}+K_{kk}-2K_{jk}\right)} \tag{5.34}$$

此时，我们结合公式(5.4)，(5.19)对公式(5.28)给出的v_j的定义稍加推导：

$$\begin{aligned} v_j &= \boldsymbol{x}_j^T\sum_{i\neq j,k}\alpha_i y_i\boldsymbol{x}_i \\ &= \boldsymbol{x}_j^T\left(\sum_{i=1}^{m}\alpha_i y_i\boldsymbol{x}_i-\sum_{i=j,k}\alpha_i y_i\boldsymbol{x}_i\right) \\ &= \left(\boldsymbol{x}_j^T\boldsymbol{\omega}+\gamma\right)-\sum_{i=j,k}\alpha_i y_i K_{i,j}-\gamma \\ &= g\left(\boldsymbol{x}_j\right)-\sum_{i=j,k}\alpha_i y_i K_{i,j}-\gamma \end{aligned} \tag{5.35}$$

同理可以推出v_k的表达式。将公式(5.35)描述的v_j, v_k的表达式以及$C=y_j\alpha_j+y_k\alpha_k$带入公式(5.34)，合并所有的$K_{kk}, K_{jj}, K_{jk}$项，可以得到：

$$\begin{aligned} \alpha_k^* &= \frac{\alpha_k\left(K_{jj}+K_{kk}-2K_{jk}\right)+y_k\left(\left(g\left(x_j\right)-y_j\right)-\left(g\left(x_k\right)-y_k\right)\right)}{\left(K_{jj}+K_{kk}-2K_{jk}\right)} \\ \alpha_k^* &= \alpha_k+y_k\frac{e_j-e_k}{\mathcal{Z}_{jk}} \end{aligned} \tag{5.36}$$

其中

$$e_i=g(\boldsymbol{x}_i)-y_i,\qquad i=j,k \tag{5.37}$$

$$\mathcal{Z}_{jk}=K_{jj}+K_{kk}-2K_{jk}=\boldsymbol{x}_j^T\boldsymbol{x}_j-2\boldsymbol{x}_j^T\boldsymbol{x}_k+\boldsymbol{x}_k^T\boldsymbol{x}_k=\left\|\boldsymbol{x}_j-\boldsymbol{x}_k\right\|^2 \tag{5.38}$$

显然$\mathcal{Z}_{jk}$是两个样本$\boldsymbol{x}_j, \boldsymbol{x}_k$之间欧氏距离的平方。对照当$y_k y_j=1$或$-1$时$\alpha_k$的两种不同取值范围（见公式(5.31)下方），当公式(5.36)中的α_k^*的数值处于α_k的取值范围内，则令$\alpha_k(t+1)=\alpha_k^*$；否则令$\alpha_k(t+1)$等于距离α_k^*最近的取值范围的边界，即$\alpha_k(t+1)=0$或$\alpha_k(t+1)=y_j C$。此后再利用公式(5.29)计算出$\alpha_j(t+1)$的数值。

$$\alpha_j(t+1)=y_j\left(C-y_k\alpha_k(t+1)\right) \tag{5.39}$$

(3) 变量选择

在前面的两个步骤中，我们假设当前循环中的 2 个乘子α_j和α_k已经被选出。但在算法实际应用中，每次迭代是选择的拉格朗日乘子对于整体优化的速度和可靠性有着重要的影响，因此我们必须设计一种合理的变量选择策略，在当前条件下选出最有利于 SMO 算法对 SVM

问题寻优的两个变量α_j和α_k。

第一个因子α_j的选择：考虑到优化过程应使那些原本不符合 KKT 条件的乘子变得服从 KKT 条件，因此应优先选择破坏 KKT 条件最严重的拉格朗日乘子进行优化。一般首先在满足$\alpha_i > 0$的乘子(对应于支撑向量)中逐个验证 KKT 条件。此时 KKT 条件要求$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) = 1$，因此首先选择破坏该 KKT 条件最严重的乘子$\alpha_j$，既：

$$j = \underset{i:\alpha_i>0}{\operatorname{argmax}}(\zeta_i) = \underset{i:\alpha_i>0}{\operatorname{argmax}}(|y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1|) \tag{5.40}$$

这里$\zeta_i, \forall i: \alpha_i > 0$是满足$\alpha_j > 0$条件的样本破坏 KKT 条件的程度。需要注意的是，根据公式(5.19)，此时的决策面方程参数$\boldsymbol{\omega}$估计为：

$$\boldsymbol{\omega} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i \tag{5.41}$$

另一个需要确定的参数为截距γ。考虑到对于所有的支撑向量，有:

$$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 = 0, \forall \alpha_i > 0 \tag{5.42}$$

因此在算法没有收敛前，每一个$\boldsymbol{x}_i$都能得到一个γ，因此一般使用公式(5.43)来估计截距γ。

$$\gamma = \frac{1}{|\mathbb{V}|}\sum_{x_i \in \mathbb{V}} (y_i - \boldsymbol{\omega}^T\boldsymbol{x}_i) \tag{5.43}$$

其中$\mathbb{V}$为所有支撑向量的集合，记为$\mathbb{V} = \{x_i | \alpha_i > 0\}$。若当前时刻$\alpha_i = 0, \forall i$，则$\mathbb{V} = \emptyset$，此时令$\gamma = 0$。

$\alpha_i = 0$的情况对应于非支撑向量样本，对应的 KKT 条件为$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) > 1$，因此$1 - y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma)$的值越大说明改样本破坏 KKT 条件约严重，则乘子α_j的选择可以写为：

$$j = \underset{i:\alpha_i=0}{\operatorname{argmax}}(\zeta_i) = \underset{i:\alpha_i=0}{\operatorname{argmax}}\big(1 - y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma)\big) \tag{5.44}$$

总的来说，用于表示 KKT 条件破坏程度的ζ_i可以写为：

$$\zeta_i = \begin{cases} |y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1|, & \alpha_i > 0 \\ 1 - y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma), & \alpha_i = 0 \end{cases} \tag{5.45}$$

ζ_i越大表示拉格朗日乘子α_i对 KKT 条件的破坏越严重。

在搜索策略上，首先借助公式(5.41)和(5.43)，对公式(5.40)的解进行逐个搜索。如果当前所有满足$\alpha_i > 0$的拉格朗日乘子均符合 KKT 条件，即满足公式(5.46)：

$$\zeta_i = |y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1| \leq \epsilon, \forall \alpha_i > 0 \tag{5.46}$$

ϵ是一个事先给定的小正数，表示验证是否符合 KKT 条件的阈值，则开始逐个搜索$\alpha_i = 0$的样本，验证其是否满足$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 \geq 0$的 KKT 条件，并根据公式(5.44)选出具有最大ζ_i

值的乘子α_j。通过上述操作可以确保从当前的所有拉格朗日乘子中找出第一个待优化拉格朗日乘子α_j。

第二个因子α_k的选择：应尽可能使得α_k在优化后得到较大的变化，进而加速整个优化进程。观察公式(5.36)可以发现:

$$\Delta\alpha_k = \alpha_k^* - \alpha_k = y_k \frac{e_j - e_k}{\mathcal{Z}_{jk}} \tag{5.47}$$

为使变化量$\Delta\alpha_k$尽可能的大，α_k的最优选择为：

$$k = \underset{i:\alpha_i>0, i\neq j}{\operatorname{argmax}}(\Delta\alpha_i) = \underset{i:\alpha_i>0, i\neq j}{\operatorname{argmax}}\left(y_i \frac{e_j - e_i}{\mathcal{Z}_{ji}}\right) \tag{5.48}$$

鉴于$\alpha_k(t+1)$的最终取值可能落在其取值范围的边界上而非α_k^*处，理论上可以对除了α_j以外的每一个拉格朗日乘子$\alpha_i, i \neq j$进行验证，找出使得α_k在单次优化中变化最大的α_k。但这样做会大幅度的增加计算量和程序的复杂度。因此通常直接使用公式(5.48)对最优的α_k进行启发式寻优。其中$\mathcal{Z}_{ji}, i \neq j$表示样本$\boldsymbol{x}_j, \boldsymbol{x}_i$之间的欧氏距离的平方，不随优化过程发生变化，因此可以先计算好并存储在一个$m \times m$的矩阵$\mathcal{Z}$中，$e_i = g(\boldsymbol{x}_i) - y_i, i = 1,2,\ldots,m$是当前判别函数的预测误差。在实际应用中，在选定一个α_j后，通常对所有的$\alpha_i, i \neq j$计算对应的$\Delta\alpha_i$，然后从大到小进行搜索，当使用当前的α_i带来的目标函数$L(\alpha_i)$取值的下降幅度ΔL（指相对上一次迭代的目标函数值的降幅）足够大时，就停止当前轮次的搜索，进入下一次迭代。采用这种启发式搜索方法可以大幅度减少寻优算法的计算复杂度。

(4) 终止条件

终止条件即要求 KKT 条件在精度ϵ内得到满足。根据公式(5.25)，可以具体转化为：

$$\begin{cases} y_i g(\boldsymbol{x}_i) - 1 \geq \epsilon, & \forall \alpha_i = 0 \\ |y_i g(\boldsymbol{x}_i) - 1| \leq \epsilon, & \forall \alpha_i > 0 \end{cases} \tag{5.49}$$

其中函数$g(\boldsymbol{x})$的参数$\boldsymbol{\omega}, \gamma$的计算，详见公式(5.41)，(5.43)。

综合上述内容，面向线性可分 SVM 问题的 SMO 算法

♦ SMO 算法步骤

输入:	训练数据$X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m) \mid \boldsymbol{x}_i \in \mathbb{R}^d, y_i \in \{+1, -1\}, i = 1,2,\ldots,m\}$ 精度ϵ
步骤:	
1	初始化$t = 0$
2	初始化拉格朗日乘子$\boldsymbol{\alpha}(t) = \{0, \ldots, 0\}$ 初始化距离矩阵$\mathcal{Z}$（公式(5.38)）
2	Repeat：
3	变量选择：$\alpha_{j(t)}, \alpha_{k(t)}$
4	变量优化：计算$\alpha_{j(t)}(t+1), \alpha_{k(t)}(t+1)$ $t = t + 1$
5	until：
6	KKT 条件在精度ϵ内得到满足（公式(5.49)）

7	根据$\boldsymbol{\alpha}(t)$，计算决策面方程参数$\boldsymbol{\omega},\gamma$（公式(5.41)，(5.43)）
输出:	参数$\boldsymbol{\omega},\gamma$

图 5-3　SMO 算法步骤

♦ 例子：基于 SMO 算法的小样本线性可分问题的 SVM 求解

设训练样本共 3 个：$\{(\boldsymbol{x}_1=[0,1]^T,y_1=-1),(\boldsymbol{x}_2=[1,0]^T,y_2=-1),(\boldsymbol{x}_3=[1,1]^T,y_1=1)\}$，精度$\epsilon=0.0001$。请利用 SMO 算法训练一个线性 SVM 分类器。

解：根据图 5-3　SMO 算法步骤配合相应的公式解答如下：

1)　初始化

首先令$t=0$；初始化$\alpha_1(t)=\alpha_2(t)=\alpha_3(t)=0$；根据公式(5.38)计算矩阵$\mathcal{Z}$，如下：

$$\mathcal{Z}=\begin{bmatrix}0&2&1\\2&0&1\\1&1&0\end{bmatrix}$$

2)　选取变量

首先计算 KKT 条件的破坏程度ζ：利用公式(5.41)，(5.43)，求出$\boldsymbol{\omega}=[0,0]^T,\gamma=0$；根据公式(5.4)，计算判别函数值：

$$g(\boldsymbol{x}_1)=\boldsymbol{\omega}^T x_1+\gamma=[0\quad 0]\begin{bmatrix}0\\1\end{bmatrix}+0=0$$

同理得$g(\boldsymbol{x}_2)=g(\boldsymbol{x}_3)=0$。带入公式(5.45)得到：

$$\zeta_1=1-y_1(\boldsymbol{\omega}^T\boldsymbol{x}_1+\gamma)=1-y_1g(\boldsymbol{x}_1)=1-(-1)\times 0=1$$

同理求得$\zeta_2=\zeta_3=1$。

其次根据 KKT 条件破坏度选择第一个待优化变量$\boldsymbol{\alpha_j}$。由于此时没有$\alpha_i(t)>0$的乘子，根据公式(5.44)，在满足$\alpha_i(t)=0$的乘子中从前向后选择ζ_i最大的一个作为α_j，得到$j(t)=1$。

最后根据预期变化值$\Delta\boldsymbol{\alpha}$选择第二个待优化变量$\boldsymbol{\alpha_k}$。利用公式(5.37)计算所有样本的预测误差：$e_1=1,e_2=1,e_3=-1$，带入公式(5.47)，得到$\Delta\alpha_2=0,\Delta\alpha_3=2$，根据公式(5.48)选择$k(t)=3$。

3)　变量寻优

首先找到$\boldsymbol{\alpha_k}$的理论最优解：

$$\alpha_{k(t)}^*=\alpha_3^*=\alpha_3+\Delta\alpha_3=0+2=2$$

其次确定$\boldsymbol{\alpha_k}$的取值范围，根据公式(5.31)的要求，先确定y_jC的数值：

$$y_{j(t)}C=y_1C=y_1(y_1\alpha_1+y_3\alpha_3)=y_1(-1\times 0+1\times 0)=0$$

考虑到$y_1y_3=-1$，根据公式(5.31)下的*情况 2*，得到当前α_k的取值范围为$[0,+\infty)$。

最后确定此轮迭代的最优解$\alpha_j(t+1),\alpha_k(t+1)$。由于$\alpha_3^*=2$在$\alpha_k$的取值范围$[0,+\infty)$内，推出$\alpha_k(t+1)=a_3^*=2$。然后将其带入公式(5.39)，得到：

$$\alpha_j(t+1)=\alpha_1(t+1)=y_1\big(C-y_3\alpha_3(t+1)\big)=-1\times(0-1\times 2)=2$$

此时可知$\alpha_1(t+1)=2,\alpha_2(t+1)=0,\alpha_3(t+1)=2$，完成了拉格朗日乘子的一次迭代优化。

4) 迭代优化

基于上述结果，进一步计算得到在$t=1$时刻，$\boldsymbol{\omega}=[2,0]^T,\gamma=-1,g(x_1)=-1,g(\boldsymbol{x}_2)=1,g(\boldsymbol{x}_3)=1$，进而可以计算得到$\zeta_1=0,\zeta_2=2,\zeta_3=0$。这意味着$\boldsymbol{x}_1,\boldsymbol{x}_3$作为支撑向量，已经完全符合 KKT 条件了，但$\boldsymbol{x}_2$作为非支撑向量，仍然不符合 KKT 条件，需要继续迭代寻优。后面的计算方法与前面的步骤 1)-3)完全相同，这里不再重复，仅将计算过程中的关键变量的阶段性数值列在表 5-1 中，供读者自己推演验证。其中。KKT 破坏度ζ一列中红色对应于$\alpha_i>0$的样本，蓝色对应$\alpha_i=0$的样本。可以看到经过 4 次迭代，破坏度$\zeta_1=\zeta_2=\zeta_3=0$，且$\alpha_i>0,\forall i=1,2,3$，说明此时所有样本均为支撑向量，且所有拉格朗日乘子均满足 KKT 条件，SVM 问题得到了完美的解决。此外，图 5-4 给出了 SVM 分类器决策面随迭代优化的变化情况，其中两个红色点表示$\boldsymbol{x}_1,\boldsymbol{x}_2$,蓝色的×表示$\boldsymbol{x}_3$，绿色直线表示决策面。从中可以看出决策面的方向和位置逐渐接近 SVM 最优解的过程。

当然在科研领域使用的 SMO 算法比我们在例题中介绍的还要复杂一些，主要实在启发式规则的运用以及搜索策略方面有一些微小的调整，但其核心思想是一致的。这意味着读者们也可以根据这一思路对基于 SMO 的 SVM 算法进行改进。

表 5-1 SVM 例题的 SMO 求解过程中的关键变量数值计算结果

t	$\boldsymbol{a}$	$\boldsymbol{\omega}$	γ	g	ζ	j	$\Delta\boldsymbol{a}$	k	a_k^*	y_jC	y_jy_k	$\alpha_k^{(+1)}$[1]	$\alpha_j^{(+1)}$
0	[0,0,0]	[0,0]	0	[0,0,0]	[1,1,1]	1	[0,0,2]	3	2	0	−1	2	2
1	[2,0,2]	[2,0]	−1	[−1,1,1]	[0,2,0]	1	[0,1,0]	2	1	2	1	1	1
2	[1,1,2]	[1,1]	−5/3	$[-\frac{2}{3},-\frac{2}{3},\frac{1}{3}]$	$[\frac{1}{3},\frac{1}{3},\frac{2}{3}]$	3	[1,1,0]	1	2	1	−1	2	3
3	[2,1,3]	[2,1]	−7/3	$[-\frac{1}{3},-\frac{1}{3},\frac{2}{3}]$	$[\frac{1}{3},\frac{2}{3},\frac{1}{3}]$	2	$[-\frac{1}{2},0,1]$	3	4	−2	−1	4	2
4	[2,2,4]	[2,2]	−3	[−1,−1,1]	[0,0,0]	–	–	–	–	–	–	–	–

[1] $\alpha_k^{(+1)}$表示$\alpha_k(t+1)$，$\alpha_j^{(+1)}$表示$\alpha_j(t+1)$

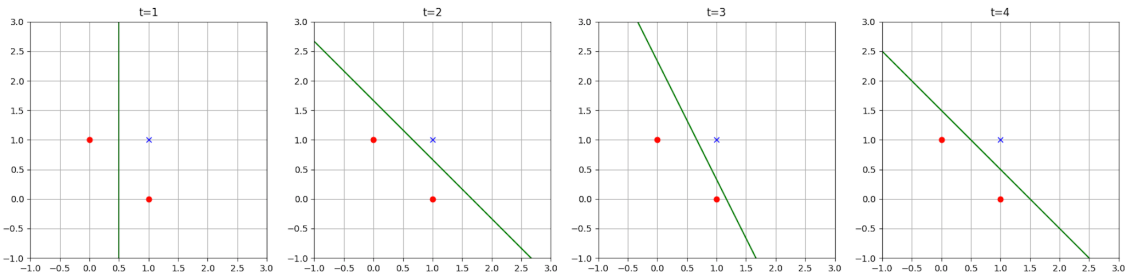

图 5-4　SMO 算法迭代优化过程中决策面变化情况

5.4　软间隔线性 SVM 方法

面向线性可分问题的 SVM 分类器学习方法是整个 SVM 算法的基础。但现实中大多数分类问题都是线性不可分的。解决这类问题有两种思路，一种思路是仍然使用线性 SVM，并允许在训练结束后仍然存在一定的错分现象，另一种思路是将线性不可分问题转化为线性可分问题再进行处理。本节主要遵循第一种思路，介绍基于“软间隔”最大化的线性 SVM 算法。

5.4.1　松弛变量的引入

对于一个线性不可分问题，要求线性 SVM 的判别函数满足$y_i(\boldsymbol{\omega}^T\boldsymbol{x_i}+\gamma)\geq 1,\forall \boldsymbol{x}_i$的约束条件（详见公式(5.13)）是不可能的。因此需要对每一个训练样本$(\boldsymbol{x}_i,y_i)$引入对应的“松弛变量”ξ_i，进而对约束条件进行如下改造

$$y_i(\boldsymbol{\omega}^T\boldsymbol{x_i}+\gamma)\geq 1-\xi_i,\qquad \xi_i\geq 0,\qquad \forall \boldsymbol{x}_i \tag{5.50}$$

公式(5.50)的意思是样本可以在松弛变量控制的程度内侵入到分类间隔区域内部。换言之，间隔区域从“坚硬不可侵入”的状态变成了“柔软可以适当侵入”的状态，因此我们称上两节介绍的算法为“硬间隔”SVM，称具有松弛变量的算法为“软间隔”SVM。

对于一个线性不可分问题，显然无法找到任何一组参数满足所有硬间隔 SVM 的约束条件，即可行解空间为空集，这显然是不可接受的。而软间隔 SVM 由于引入了松弛变量，从而放宽了约束条件，因此可以解决线性不可分问题。

5.4.2　软间隔最大化问题

虽然引入松弛变量可以方框 SVM 优化问题的约束条件，但松弛变量ξ_i应尽可能的小以减轻样本侵入软间隔区域的程度。所以需要在原有的硬间隔 SVM 目标函数（见公式(5.14)）基础上，加入与松弛变量有关的代价项。则软间隔 SVM 对应的优化问题可以写为：

(5.51)

$$m$$

$$\xi_i \geq 0, \qquad i = 1,2,\dots,m$$

在目标函数方面，$C > 0$称为惩罚因子，C越大则模型对于样本侵入软间隔区域的容忍度就越低。与原有的目标函数对比可以看出，软间隔 SVM 一方面希望间隔越大越好，一方面希望侵入软间隔的情况越少越好。在线性不可分问题中，这两种情况属于“鱼和熊掌不可得兼”，所以惩罚因子C在其中起到了一个调和矛盾的作用。

在约束条件方面，松弛变量ξ_i的引入使得模型对决策面参数$\boldsymbol{\omega}, \gamma$的要求适当降低了。但这种降低通常是以包含松弛变量$\xi_i$的目标函数数值的增加为代价的。换言之，松弛变量$\xi_i$起到了调和目标函数与约束条件之间矛盾的作用。

5.4.3 软间隔 SVM 的求解

相比于硬间隔 SVM，软间隔 SVM 的优化问题的额外增加了 m 个自变量：$\xi_i, i = 1,2,\dots,m$，以及对应的 m 个线性不等式约束条件$\xi_i \geq 0, i = 1,2,\dots,m$，但其数学形式仍然是一个线性不等式约束条件下的凸二次优化问题，可以采用拉格朗日对偶方法加以求解。

首先基于公式(5.51)描述的原始优化问题，利用拉格朗日乘子法，构造拉格朗日函数

$$L(\boldsymbol{\omega}, \gamma, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{m}\xi_i - \sum_{i=1}^{m}\alpha_i(y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i) - \sum_{i=1}^{m}\beta_i\xi_i \tag{5.52}$$

其中$\alpha_i \geq 0, \beta_i \geq 0, \forall i = 1,2,\dots,m$。则原始问题为：

$$\min_{\boldsymbol{\omega},\gamma,\boldsymbol{\xi}}\left[\max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i\geq 0,\beta_i\geq 0} L(\boldsymbol{\omega}, \gamma, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})\right] \tag{5.53}$$

相应的拉格朗日对偶问题为：

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}:\alpha_i\geq 0,\beta_i\geq 0}\left[\min_{\boldsymbol{\omega},\gamma,\boldsymbol{\xi}} L(\boldsymbol{\omega}, \gamma, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})\right] \tag{5.54}$$

首先求$L(\boldsymbol{\omega}, \gamma, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta})$对$\boldsymbol{\omega}, \gamma, \boldsymbol{\xi}$的极小值，令相应的偏导数为 0，得到方程：

$$\frac{\partial L}{\partial \boldsymbol{\omega}} = \mathbf{0} \Longrightarrow \boldsymbol{\omega} = \sum_{i=1}^{m}\alpha_i y_i \boldsymbol{x}_i \tag{5.55}$$

$$\frac{\partial L}{\partial \gamma} = 0 \Longrightarrow 0 = \sum_{i=1}^{m}\alpha_i y_i \tag{5.56}$$

$$\frac{\partial L}{\partial \boldsymbol{\xi}} = \mathbf{0} \Longrightarrow C - \alpha_i - \beta_i = 0, \forall i = 1,2,\dots,m \tag{5.57}$$

将公式(5.55), (5.56)和(5.57)带入公式(5.52)，得到：

$$\min_{\boldsymbol{\omega},\gamma,\boldsymbol{\xi}} L(\boldsymbol{\omega}, \gamma, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j \tag{5.58}$$

再根据拉格朗日对偶法，求$\min_{\boldsymbol{\omega},\gamma,\boldsymbol{\xi}} L(\boldsymbol{\omega},\gamma,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\beta})$对$\boldsymbol{\alpha}$和$\boldsymbol{\beta}$的最大值。需要注意的是，由于公式(5.57)的约束，α_i和β_i只相当于一个有效变量，因此$\min_{\boldsymbol{\omega},\gamma,\boldsymbol{\xi}} L(\boldsymbol{\omega},\gamma,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\beta})$中并不包含变量$\beta_i$，也就不需要对$\beta_i$求极大值，因此最终的对偶问题写为：

$$\begin{aligned} &\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j \\ &s.t. \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \\ &\qquad C - \alpha_i - \beta_i = 0 \\ &\qquad\qquad \alpha_i \geq 0, i = 1,2,\dots,m \\ &\qquad\qquad \beta_i \geq 0, i = 1,2,\dots,m \end{aligned} \tag{5.59}$$

利用α_i和β_i之间的等式约束关系，将约束条件中的$\beta_i \geq 0$消掉，可以得到关于α_i的更简单的约束形式：

$$0 \leq \alpha_i \leq C, i = 1,2,\dots,m \tag{5.60}$$

由于增加了 m 个自变量$\xi_i, i = 1,2,\dots,m$和对应的 m 个拉格朗日乘子$\beta_i, i = 1,2,\dots,m$，软间隔 SVM 优化问题的 KKT 条件发生了一些变化，具体如下：

$$\begin{cases} \alpha_i \geq 0 & (1) \\ y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i \geq 0 & (2) \\ \alpha_i(y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i) = 0 & (3) \\ \beta_i \geq 0 & (4) \\ \xi_i \geq 0 & (5) \\ \beta_i\xi_i = 0 & (6) \end{cases}, i = 1,2,\dots,m \tag{5.61}$$

基于上述 KKT 条件，可以同样采用 SMO 算法对软间隔 SVM 问题进行求解。相比于硬间隔 SVM，软间隔 SVM 增加了$\alpha_i \leq C$的约束条件，这一点可以在 SMO 算法变量寻优阶段通过增加一个范围边界的方式加以解决。

5.4.4 软间隔 SVM 的几何解释

在推导 SVM 分类器的决策面方程参数$\boldsymbol{\omega},\gamma$的最优解与拉格朗日乘子$\alpha_i$的关系时（公式(5.41)和(5.42)），我们注意到$\boldsymbol{\omega},\gamma$仅由满足$\alpha_i > 0$条件的样本决定，也就是说只有支撑向量对于 SVM 分类器的建立起到了实际作用（这也是它们被称为支撑向量的原因）。对于硬间隔 SVM，这种支撑作用在几何层面上很好理解，即支撑向量样本点正好处于分类间隔区域的边界上。但对于软间隔 SVM 而言，样本与决策面以及间隔区域的关系相对复杂。图 5-5 和表 5-2 对不同类型的样本做出了对照说明。

当$\alpha_i = 0$时，根据公式(5.57)：$C - \alpha_i - \beta_i = 0$，有$\beta_i = C$；再根据 KKT 条件(6)：$\beta_i\xi_i = 0$，有$\xi_i = 0$；再根据 KKT 条件在$\alpha_i = 0$时$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i > 0$，推出$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) > 1$。也就是说，此时样本$\boldsymbol{x}_i$在间隔区域外，对应于 **Case1**。

剩余满足$\alpha_i > 0$条件的样本，根据 KKT 条件(3)，必然有$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i = 0$；由于$\xi_i \geq 0$，所以有$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) \leq 1$，说明样本$\boldsymbol{x}_i$在间隔区域边界上或边界内，$\boldsymbol{x}_i$是支撑向量，会对最终的决策面方程参数的确定起到影响，对应于 **Case2**。

在 Case2 中，α_i仍有两种可能性。一是当$0 < \alpha_i < C$时，根据 KKT 条件(3)首先有$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i = 0$；根据$C - \alpha_i - \beta_i = 0$，有$\beta_i > 0$；进而根据 KKT 条件(6)：$\beta_i\xi_i = 0$，有$\xi_i = 0$。因此$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) = 1$，说明样本$\boldsymbol{x}_i$在间隔区域的边界上，没有侵入间隔区域内部，对应 **Case2-1**；二是当 $\alpha_i = C$时，可以推出$\beta_i = 0$，因此$\xi_i > 0$，进而推出$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) < 1$，说明样本$\boldsymbol{x}_i$已经越过了间隔区域的边界，对应 **Case2-2**。

在 Case2-2 中，根据ξ_i的取值不同仍有两种情况。第一种情况是当$0 < \xi_i < 1$时，$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) = 1 - \xi_i$推出$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) > 0$，说明样本点仍然在决策面的正确的一侧，即样本$\boldsymbol{x}_i$仍可以被正确分类分类，对应 **Case2-2-1**。第二种情况是当$\xi_i \geq 1$时，$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) \leq 0$，样本点$\boldsymbol{x}_i$到达或越过了决策面，会被错误分类，对应 **Case2-2-2**。

通过上述分析，不难理解松弛变量ξ_i的几何含义，定性地看，ξ_i描述的是样本侵入软间隔区域的程度。$\xi_i = 0$且$\alpha_i = 0$表示样本没有侵入间隔区域；$\xi_i = 0$且$\alpha_i > 0$表示样本在间隔区域边界上；$0 < \xi_i < 1$表示样本在间隔区域边界与决策面之间；$\xi_i = 1$表示样本在决策面上；$\xi_i > 1$表示样本越过了决策面，被错误分类。定量地看，对于所有的支撑向量（满足$\alpha_i > 0$条件的样本），根据 KKT 条件(3)，有$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 = \xi_i$，因此其样本$\boldsymbol{x}_i$到自己类别对应一侧的间隔区域边界直线$y_i(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) - 1 = 0$的距离为：

$$d(\boldsymbol{x}_i) = \frac{y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1}{\|\boldsymbol{\omega}\|} = \frac{\xi_i}{\|\boldsymbol{\omega}\|} \tag{5.62}$$

而间隔区域宽度的一半为$1/\|\boldsymbol{\omega}\|$。

根据上面的几何解释，对照公式(5.51)所描述的优化问题，不难理解软间隔 SVM 就是希望对决策面的优化实现两个目标：1) 最大化间隔宽度，2) 最小化样本侵入间隔区域的平均距离。但这两个目标显然是相互矛盾的，间隔区域宽度越宽，侵入间隔区域的样本数量就越多，平均侵入距离就越大。因此算法设计了一个惩罚因子 C 来调和两种相互矛盾的目标。

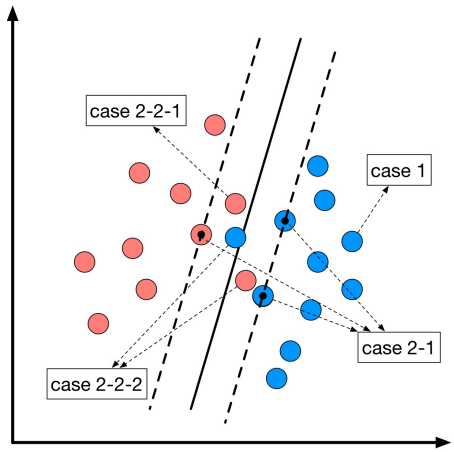

图 5-5　软间隔 SVM 中不同拉格朗日乘子对应的样本点位置分布情况示意图。

表 5-2　SVM 例题的 SMO 求解过程中的关键变量数值计算结果

<table>
<tr><td rowspan="4">样本点</td><td colspan="3">Case1：
$\alpha_i = 0$,
$\boldsymbol{x}_i$在间隔区域外</td></tr>
<tr><td rowspan="3">Case2：
$\alpha_i > 0$,
$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) = 1 - \xi_i$</td><td colspan="2">Case2-1:
$\alpha_i < C$,
$\beta_i > 0$, $\xi_i = 0$,
$\boldsymbol{x}_i$在间隔区域边界上。</td></tr>
<tr><td>Case2-2:
$\alpha_i = C$,
$\beta_i = 0, \xi_i > 0$</td><td>$\xi_i < 1$,
$\boldsymbol{x}_i$在间隔区域内正确的一侧；
$\boldsymbol{x}_i$能被正确分类。</td></tr>
<tr><td>$\xi_i \geq 1$,
$\boldsymbol{x}_i$越过了决策面或在决策面上；
$\boldsymbol{x}_i$被错误分类。</td></tr>
</table>

5.4.5　经验风险与结构化风险

有别于软间隔 SVM 的几何解释，我们还可以公式(5.51)目标函数的代数形式出发，去重新解读软间隔 SVM 的意义。可以对优化问题进行改造如下：

$$\min_{\boldsymbol{\omega},\gamma} \sum_{i=1}^{m} \ell_h(\boldsymbol{x}_i) + \lambda\|\omega\|^2 \tag{5.63}$$

其中

$$\ell_h(\boldsymbol{x}) = \begin{cases} 1 - y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma), & y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) < 1 \\ 0, & y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) \geq 1 \end{cases} \tag{5.64}$$

$$\lambda = \frac{1}{2C} \tag{5.65}$$

令$\ell_h(\boldsymbol{x}_i) = \xi_i$，根据公式(5.64)，很容易推出$\xi_i \geq 0$，且$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i \geq 0$。因此，公式(5.63)描述的新的无约束优化问题与公式(5.51)描述的有约束优化问题，只是在目标函数值上差了一个惩罚因子C，其他方面完全等价。新问题的解就是软间隔 SVM 问题的解。

我们称函数$\ell_h(\boldsymbol{x})$为**合页损失函数(Hinge loss function)**，如果将$y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma)$替换为$x'$，则$\ell_h(\boldsymbol{x})$的函数曲线如所示。从中可以看出合页损失函数是一个分段线性函数，用于反映分类结果的代价。当$x' > 1$时，样本$\boldsymbol{x}$在间隔区域以外，将会被正确分类，因此代价为 0；当$x' \leq 1$时，样本$\boldsymbol{x}$开始侵入间隔区域，甚至可能错误分类$x' \leq 0$，因此会产生分类代价$1 - x'$。读者可能会有疑问，按照分类问题的设定，分类损失应该是一个 0-1 阶跃函数，也就是：

$$\ell_b(\boldsymbol{x}) = \begin{cases} 1, & y(\boldsymbol{\omega}^T\boldsymbol{x}+\gamma) \le 0 \\ 0, & y(\boldsymbol{\omega}^T\boldsymbol{x}+\gamma) > 0 \end{cases} \tag{5.66}$$

事实上**错误!未找到引用源。**节介绍的感知器就采用了$\ell_b(\boldsymbol{x})$作为损失函数。然而 0-1 阶跃函数$L_b(\boldsymbol{x})$存在两个问题。一是在代数形式上，0-1 损失函数在$y(\boldsymbol{\omega}^T\boldsymbol{x}+\gamma)=0$处（也就是决策面上）不是连续可导的，因此不利于学习算法的梯度优化；二是只考虑了训练样本的定性分类结果，没有考虑对错的程度带来的影响，因此对于测试样本的泛化能力不够好。而合页损失函数$\ell_h(\boldsymbol{x})$可以视为 0-1 损失函数$\ell_b(\boldsymbol{x})$的上界，不但处处连续可导，而且由于考虑了分类对错在程度上的差别，因此分类结果的置信度更高，不容易发生欠拟合现象。我们将此类原始代价函数的上界损失函数，称为**代理损失函数(Surrogate loss function)**。如何设计一款好的代理损失函数，是机器学习领域的核心问题之一。

目标函数中的另外一项$\|\omega\|^2$，来自于 SVM 问题假设中的间隔最大化思想，但对于一个通用的分类器模型而言，它还有一个更常用的称谓——**正则化项(Regularization term)**。正则化是机器学习领域中一种常用的提升模型泛化能力，减轻过拟合现象的技术手段。对于一个线性可分问题而言，存在无穷多个决策面能够将所有训练样本正确分类，这与我们在线性回归模型讨论的过拟合的情况非常相似，即可以找到无穷多个平面穿过所有的样本点。然而这无穷多个最优解的泛化能力显然不同，SVM 认为具有最大分类间隔的最优解应具有最好的泛化能力。由于$\|\omega\|$与分类间隔成反比，最小化$\|\omega\|$相当于最大化分类间隔，因此可以在总的目标函数中加入$\|\omega\|^2$项，以提升模型的泛化能力。关于正则化的另一种解释是：“在所有能够解决当前训练数据集上的分类问题的模型假设中，我们应该选择其中最简单的一个假设。”这一论断在科学方法论中被称为**“奥卡姆剃刀原理(Occam’s Razor)”**（详见**错误!未找到引用源。**节）。在我们这个问题中具体表现为：“在能够解决训练数据的模式识别问题的前提下，参数向量$\boldsymbol{\omega}$的非零元素个数和数值越小，预期的泛化能力越好。”因此对$\|\omega\|^2$的最小化有助于提升模型的泛化能力。

综上所述，合页损失函数$\ell_h(\boldsymbol{x})$的主要目的是减少模型在训练数据上的经验误差，避免欠拟合现象的发生，因此又称为**经验风险(Empirical risk)**；正则化项$\|\omega\|^2$的主要目的是减少模型在测试数据上的泛化误差，减轻过拟合现象，由于测试数据未知，$\|\omega\|^2$仅与模型本身的结构和参数有关，因此又称为**结构化风险(Structural risk)**。对于一个机器学习模型，这两种风险都非常重要，但同时又是相互矛盾的。因为对于一个线性不可分问题来说，并不存在一个理想的线性分类器能够使得经验风险为零，因此分类器越复杂越有助于减少经验误差，而同时越复杂的分类器的结构化风险就越大。因此软间隔 SVM 模型采用一个系数λ来调和经验风险和结构化风险之间的矛盾。

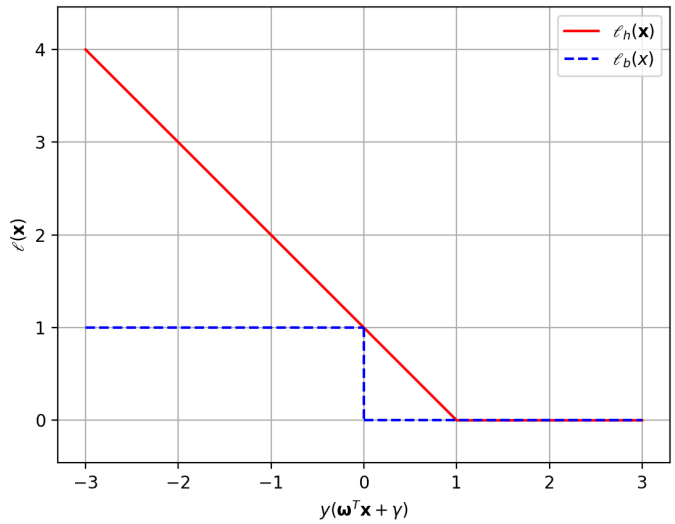

图 5-6　合页损失函数$\ell_h(\boldsymbol{x})$与 0-1 阶跃损失函数$\ell_b(\boldsymbol{x})$曲线图

5.5　非线性支持向量机

虽然软间隔 SVM 为线性不可分问题提供了一个解决思路，但由于仍然采用了线性模型的基本假设，因此软间隔 SVM 无法将所有训练样本都正确分类，也就无法为线性不可分问题提供完美的解决方案。针对这一不足，本节将在 SVM 模型架构基础上介绍一种全新的解决思路，通过将线性不可分问题转化为线性可分问题，再引入核技巧实现问题的求解。通过这种方法学得的 SVM 模型在原始特征空间中将具有非线性决策面，因此称为“非线性 SVM”。

5.5.1　非线性映射

按照一般的思维习惯，解决线性不可分问题的思路是将判别函数$g(\boldsymbol{x})$从线性转化为非线性，此时决策面方程$g(\boldsymbol{x})=0$是一个非线性方程，决策面是特征空间中的一个超曲面。我们希望这个超曲面能够将训练数据集中的两类样本完美的分开。但沿着这个思路去设计非线性分类器存在几个难题：

1) 判别函数$g(\boldsymbol{x})$的非线性函数形式与参数应该如何设定？是使用多项式、高斯函数还是三角函数、指数或是对数函数？其中部分函数的阶数——例如多项式和三角函数——该如何选定？我们选择的非线性函数形式是否适合当前特征空间中的样本分布情况？
2) 线性判别函数假设下的 SVM 问题的求解已经如此复杂，如果采用了非线性判别函数，SVM 优化问题很可能不再是线性约束条件下的凸优化问题？之前学习的 KKT 条件和拉格朗日对偶等技巧都会失效。以我们目前掌握的数学技巧将很难找到这个优化问题的理论上的最优解

尽管存在上述问题，但这种思路并非不可行，事实上神经网络模型采用的就是这种思路。但如果我们硬要将这种思路与 SVM 算法框架结合在一起，问题就会变得难以解决。所以 SVM 算法采用了一种截然不同的思路——非线性映射+核技巧——巧妙地解决了这个问题。

该思路将训练数据映射到一个新的特征空间中，使得原本的线性不可分问题在新的特征空间中变为线性可分问题，之后再使用线性 SVM 加以解决。根据**错误!未找到引用源。**节介绍的线性代数基础理论，如果一个数据集在原有的特征空间$\mathcal{X}$中是不可分的，那么在$\mathcal{X}$的任意线性变换或$\mathcal{X}$的任意线性子空间中，这个数据集仍然是不可分的。所以我们要寻找的新特征空间必然是原特征空间$\mathcal{X}$的一个**非线性映射**。关于这一点，我们可以用一个具体的例子以形象的说明。

如图 5-7 (a)所示，红色点和蓝色“+”号分别对应于二维空间$\mathcal{X} = \mathrm{x}_1 \times x_2$中的两类，此处$\mathrm{x}_1, \mathrm{x}_2$分别表示样本特征的第一个维度与第二个维度。现在我们采用非线性映射的方式增加一个新的维度$\mathrm{x}_3 = \mathrm{x}_1^2 + \mathrm{x}_2^2$，并将两类样本映射到新构造的三维空间$\mathcal{H} = \mathrm{x}_1 \times \mathrm{x}_2 \times \mathrm{x}_3$中，其分布如图 5-7 (b)所示。显然训练数据在原来的二维空间中是线性不可分的，但经过简单的非线性映射，在新构造的三维空间中就变成了一个非常简单的线性可分问题，只需要构造一个$\mathrm{x}_3 = 1$的决策平面，就可以完美解决原始数据的分类问题。

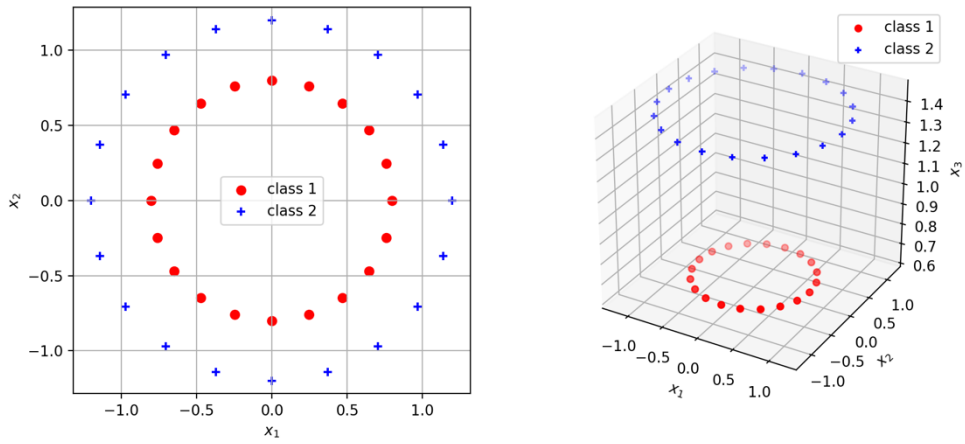

图 5-7　数据非线性映射举例示意图

将这个从二维空间到三维空间的映射记为函数$\boldsymbol{\phi}(\boldsymbol{x})$，定义如下：

$$\boldsymbol{\phi}(\boldsymbol{x}): \mathcal{X} \to \mathcal{H} \tag{5.67}$$

其中$\mathcal{X}$是样本$\boldsymbol{x}$对应的随机变量 x 的特征希尔伯特空间，$\mathcal{H}$是令一个希尔伯特空间。借助非线性映射$\boldsymbol{\phi}$，表面上看线性不可分问题似乎已经被非线性映射问题解决了，但实际上我们如何找到一个非线性映射函数$\boldsymbol{\phi}(\boldsymbol{x})$确保线性不可分数据经过映射后变成线性可分，是一个非常棘手的问题。相比于前面提到的非线性判别函数$g(\boldsymbol{x})$的设计问题，非线性映射面临的困境是相似的。而“核技巧”的出现，为非线性 SVM 的求解指出了一条非常“天才”的解决思路。

5.5.2 核化 SVM

先不去考虑非线性映射函数$\boldsymbol{\phi}(\boldsymbol{x})$的具体数学形式，仅从$\boldsymbol{\phi}(\boldsymbol{x})$所在的高维空间中的软间隔 SVM 模型出发，参考公式(5.59)和(5.60)，写出对应的优化问题如下：

$$\begin{aligned} &\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) \\ &s.t. \qquad \sum_{i=1}^{m} \alpha_i y_i = 0 \\ &\qquad\quad 0 \le \alpha_i \le C, i = 1,2,\ldots,m \end{aligned} \tag{5.68}$$

其中

$$K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) = \phi(\boldsymbol{x}_i)^T \boldsymbol{\phi}\left(\boldsymbol{x}_j\right), \forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \Omega(\mathbf{x}) \tag{5.69}$$

函数$K(\ \ )$称为核函数，$\boldsymbol{\phi}(\boldsymbol{x})$是对应的映射函数。显然，公式(5.27)中的核函数可以看出是公式(5.69)定义的核函数在映射函数$\boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{x}$条件下的特例。

观察公式(5.69)可以发现，核函数的输出是两个向量的内积，因此是一个标量。从形式上看，对于一个给定的映射$\boldsymbol{\phi}(\boldsymbol{x})$，有且只有一个核函数$K(\ \ )$与之对应；但反之则不然，对于一个确定的核函数$K(\ \ )$，却可能存在多个甚至无穷个映射$\boldsymbol{\phi}(\boldsymbol{x})$能够与之对应，而$\boldsymbol{\phi}(\boldsymbol{x})$对应的希尔伯特空间$\mathcal{H}$的维度也存在多种甚至无穷多种可能性，理论上，空间$\mathcal{H}$甚至可以为无穷维。

参考 SMO 算法可以发现，在 SVM 问题求解过程中，样本$\boldsymbol{x}$总是与模型参数$\boldsymbol{\omega}$同时以$\boldsymbol{\omega}^T\boldsymbol{x}$的形式出现的。结合公式(5.54)，用映射后的$\boldsymbol{\phi}(\boldsymbol{x})$取代原特征空间中的样本$\boldsymbol{x}$，则有

$$\boldsymbol{\omega}^T \phi(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i y_i \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) \tag{5.70}$$

这意味着非线性映射$\boldsymbol{\phi}(\boldsymbol{x})$在整个 SVM 问题的求解过程中，从未单独出现过，而总是通核函数$K(\ \ )$的形式出现，因此我们并不需要具体知道$\boldsymbol{\phi}(\boldsymbol{x})$的具体数学形式，只需要能够对任意样本$\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$，计算出他们的核函数值$K\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$即可。到这里，读者可以回头再简单的浏览 SMO 算法的计算过程，可以看到计算过程与样本$\boldsymbol{x}$相关的项，均写成了核函数K的形式，因此完全可以利用重新定义的核函数$K(\ \ )$在 SMO 算法的框架下对非线性 SVM 问题进行求解，这种基于核函数的 SVM 模型被称为核化 SVM。

5.5.3 核函数的判定与选择

根据公式(5.69)，核函数$K(\boldsymbol{x}, \boldsymbol{y})$必须能够写成非线性映射$\boldsymbol{\phi}(\boldsymbol{x})$和$\boldsymbol{\phi}(\boldsymbol{y})$的内积，然而在实际使用中通常并不会给出$\boldsymbol{\phi}(\boldsymbol{x})$的显式表达式，那么该如何判断一个函数$K(\boldsymbol{x}, \boldsymbol{y})$是不是核函数呢？对于 SVM 而言，这关系到如何设计或选择核函数$K(\boldsymbol{x}, \boldsymbol{y})$的数学形式的问题。

从定义出发，函数$K(\boldsymbol{x}, \boldsymbol{y})$是核函数的充分必要条件是：

1） $K(\boldsymbol{x}, \boldsymbol{y})$可以写成映射函数$\boldsymbol{\phi}(\boldsymbol{x})$和$\boldsymbol{\phi}(\boldsymbol{y})$的内积；

2） 映射函数$\phi(\boldsymbol{x})$能够将原向量 x 映射到一个新的希尔伯特空间。

由于上述两个条件在实际应用中很难验证，可以将其转化为以下的核函数判定条件：

定理 5.1（核函数判定）：令$\mathcal{X}$为输入空间，函数K是定义在$\mathcal{X}\times\mathcal{X}$上的对称函数，当且仅当对于任意数据$D=\{\boldsymbol{x}_i\in\mathcal{X}|i=1,2,\dots,m\}$, 函数$K$的 Gram 矩阵均为半正定时，函数 K 是核函数。

这里函数K的 Gram 矩阵$\mathcal{K}$定义为：

$$\mathcal{K}=\begin{bmatrix}K(\boldsymbol{x}_1,\boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_1,\boldsymbol{x}_m)\\ \vdots & \ddots & \vdots\\ K(\boldsymbol{x}_m,\boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_m,\boldsymbol{x}_m)\end{bmatrix} \tag{5.71}$$

关于定理 5.1 的证明应分别从核函数的定义即$K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)$推出 Gram 矩阵$\mathcal{K}$的半正定性，再反向从$\mathcal{K}$的正定性推出从原空间$\mathcal{X}$到另一个希尔伯特空间$\mathcal{H}$的映射$\phi(\boldsymbol{x})\in\mathcal{H}$的存在性。满足上述充分必要条件的核函数$K$又被称为**正定核**或**再生核**，其对应的映射空间$\mathcal{H}$被称为**再生希尔伯特空间**。由于上述证明过程设计较多的线性代数与泛函分析概念和定义，因此在本节不做详细介绍，具体内容建议查阅李航老师的《统计学习方法》第二版。

即便有了定理 5.1 的帮助，但由于涉及到基于任意数据 D 的 Gram 矩阵的无限性问题，要验证一个函数是否是正定核函数仍然十分困难。因此在实际应用中，我们通常使用一些已经被证明的常用核函数，具体包括：

1） 多项式核函数（Polynomial kernel function）

$$K(\boldsymbol{x},\boldsymbol{z})=(\boldsymbol{x}^T\boldsymbol{z}+1)^q, q\geq 1 \tag{5.72}$$

根据公式(5.4)和(5.70)，判别函数$g(\boldsymbol{x})$可以写为：

$$g(\boldsymbol{x})=\boldsymbol{\omega}^T\boldsymbol{x}+\gamma=\sum_i^m \alpha_i^* y_i\left(\boldsymbol{x}_i^T\boldsymbol{x}+1\right)^q+\gamma^* \tag{5.73}$$

其中$\alpha_i^*, i=1,2,\dots,m$为每一个样本对应的拉格朗日乘子在 SVM 问题中的最优解，γ^*为 SVM 求解得到的决策面方程常数项的最优值。

2） 高斯核函数（Gaussian kernnel function）

$$K(\boldsymbol{x},\boldsymbol{z})=\exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{z}\|^2}{2\sigma^2}\right), \sigma^2>0 \tag{5.74}$$

对应的判别函数为：

$$g(\boldsymbol{x})=\sum_i^m \alpha_i^* y_i \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}{2\sigma^2}\right)+\gamma^* \tag{5.75}$$

3） Sigmoid 核函数（Sigmoidal kernel function）

$$K(\boldsymbol{x},\boldsymbol{z})=\tanh(\beta\boldsymbol{x}^T\boldsymbol{z}+\theta), \beta,\theta>0 \tag{5.76}$$

对应的判别函数为：

$$g(\boldsymbol{x}) = \sum_{i}^{m} \alpha_i^* y_i \exp\left(\tanh\left(\beta \boldsymbol{x}_i^T \boldsymbol{x} + \theta\right)\right) + \gamma^* \tag{5.77}$$

从以上三种核函数对应的判别函数中我们可以更加形象的理解非线性 SVM 算法的本质。在判别函数的连加项中，只有$\alpha_i \neq 0$的样本$\boldsymbol{x}_i$，也就是支撑向量，才会对最终的判别函数值产生影响。支撑样本对最终判别函数的影响包含“基本度量”与“非线性关系”两个方面的因素。基本度量是描述当前测试样本$\boldsymbol{x}$和某一个支撑样本$\boldsymbol{x}_i$之间的相似性关系的基本定义，例如多项式核与 Sigmoid 核都采用了内积$\boldsymbol{x}_i^T \boldsymbol{x}$作为基本度量，内积值越大表示样本$\boldsymbol{x}$与$\boldsymbol{x}_i$的相似性越大；而高斯核采用了欧氏距离$\|\boldsymbol{x} - \boldsymbol{x}_i\|$作为基本度量，距离越小表示样本$\boldsymbol{x}$与$\boldsymbol{x}_i$的相似性越大。在基本度量的基础上，不同的核函数选择的非线性关系也有所差别，分别表现为多项式函数、高斯函数和反切函数。对于线性 SVM 而言，支撑样本$\boldsymbol{x}_i$对于当前测试样本$\boldsymbol{x}$的判别函数$g(\boldsymbol{x})$的贡献与基本度量$\boldsymbol{x}_i^T \boldsymbol{x}$成线性关系；但对于多项式核化 SVM 而言，支撑样本$\boldsymbol{x}_i$对最终决策的贡献与基本度量$\boldsymbol{x}_i^T \boldsymbol{x}$值呈多项式关系。这意味着$\boldsymbol{x}_i^T \boldsymbol{x}$值也大，支撑样本$\boldsymbol{x}_i$的相对作用也越大。

随着基本度量$\boldsymbol{x}_i^T \boldsymbol{x}$的线性变化，支撑样本对于最终决策的贡献比例在进行非线性变化。总体上这种变化表现为支撑样本与测试样本相似性越高，则该支撑样本对于最终决策的贡献比例就越大。

核化 SVM 通过核技巧与升维映射的思想可以将低维空间中的线性不可分问题尽量转化为高维空间中的线性可分问题并加以求解。如果使用了核技巧后依旧不能将所有的训练样本完全正确分类，还可以进一步使用软间隔算法得到一个相对较好的分类结果。在很多实战问题中，软间隔技术的使用还有利于提升模型的鲁棒性与泛化能力。总之，通过对核函数与软间隔算法的合理组合得到的非线性 SVM 模型可以广泛应用于各类复杂的高维数据分类问题的求解。

本章思维导图

本章习题