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，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思想上再前进一步，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？支持向量机算法对上述思路进行了卓有成效的探索与实践。

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**，主要用于解决分类任务，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，红色和蓝色样本分属两类，他们显然可以被一条直线分开的，这种分类问题被称为**线性可分问题(Linear Separable Problem)**。然而能够将两类样本分开的直线显然不止一条。图 5-1 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，对应于一个线性分类器。对于图中已有样本，A 和 B 的分类结果是一样的，但如果考虑到还未观测到的潜在数据，两者的分类性能可能存在差别。

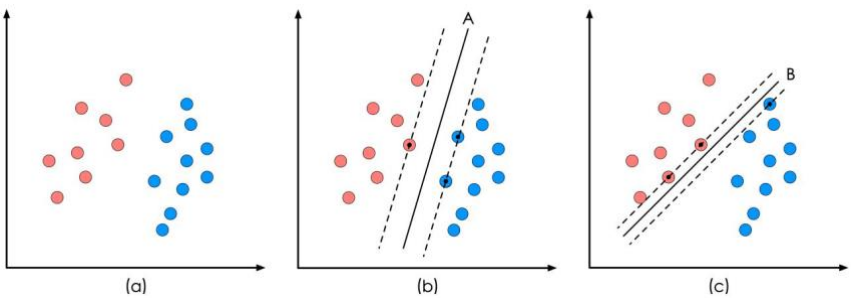


图 5-1　二分类问题描述

SVM 算法认为分类器 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 SVM 独有的一个概念——**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方向和距离决策面最近的几个样本决定。此时两条平行虚线所夹区域的中轴线可以作为该方向对应的决策面，两条虚线之间的距离称为该决策面的分类间隔。显然每一个可能把数据集正确分开的方向都对应一个决策面和一个分类间隔，而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正确地分开。不同方向的决策面在分类间隔上往往是不同的，其中具有**最大间隔(Maximal Margin)**的决策面就是 SVM 的最优解。距离最优决策面最近的样本（两侧虚线穿过的点），称为 SVM 的**支持向量(Support Vector)**。

SVM 算法的目标是找到具有最大间隔的决策面。这个优化问题的自变量表面上看似乎是决策面的方向和位置参数，但经过漫长的公式推导后会发现这些参数可以被约减掉，最终的分类间隔只取决于支持向量的选择。这一点从图 5-1 中也可以理解。

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，本章需要介绍如何把 SVM 变成用数学语言描述的最优化问题模型，这里将会利用第二章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。建议读者在继续阅读之前应先结合5.2 节 SVM 的模型，重温本书 2.3 节的相关内容。

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基本要素：1）优化对象，也就是自变量，即期望通过改变哪些因素来使目标函数达到最优；2）目标函数，期望达到最优的指标；3）约束条件，任务要求最优解必须满足的某些条件。在线性 SVM 算法中，目标函数显然是“分类间隔”，优化对象是线性决策面方程的参数，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。

5.2.1 自变量——决策面方程

在第四章线性模型中，我们已经给出了线性决策面方程的一般描述，即:

$$\boldsymbol{w}^T\boldsymbol{x} + w_0 = 0 \tag{5.1}$$

其中$\boldsymbol{w} = [w_1, w_2, \ldots, w_d] \in \Re^d, w_0 \in \Re$。在本章，为了在后续的推导中能够更加清楚的区分参数向量$\boldsymbol{w}$与表示常数项的参数w_0，令$\boldsymbol{\omega} = \boldsymbol{w}, \gamma = w_0$，则线性决策面方程可以写为：

$$g(\boldsymbol{x}) = \boldsymbol{\omega}^T\boldsymbol{x} + \gamma = 0 \tag{5.2}$$

其中，$g(\boldsymbol{x})$是二分类问题的判别函数。根据第四章的内容可知向量$\boldsymbol{\omega}$垂直于决策面$g(\boldsymbol{x}) = 0$，也就是说决策面方向是由向量$\boldsymbol{\omega}$控制的。γ是截距，它控制了决策面的位置。由于 SVM 算法的任务是找到一个最优的线性分类器，即某个线性决策面，所以优化问题的自变量就是决策面方程的参数$\boldsymbol{\omega}$和γ。

5.2.2 目标函数——分类“间隔”

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到决策面的距离的二倍，如图 5-2 所示。因此分类间隔W，可以基于点$\boldsymbol{x}_s$到决策面$g(\boldsymbol{x}) = 0$的垂直距离计算公式得到，如公式(5.3)所示。

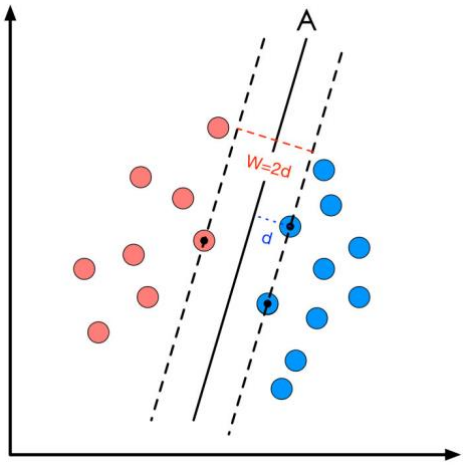


图 5-2　分类间隔计算

$$W = 2d = \frac{2|\boldsymbol{\omega}^T \boldsymbol{x}_s + \gamma|}{\|\boldsymbol{\omega}\|} \tag{5.3}$$

其中$\|\boldsymbol{\omega}\|$是向量$\boldsymbol{\omega}$的模，$\boldsymbol{x}_s$表示支持向量样本。整个 SVM 问题的核心就是寻找一组满足约束条件的$\boldsymbol{\omega}$和γ，使得公式(5.3)对应的目标函数最大化。

5.2.3 约束条件——样本正确分类

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，所以需要讨论一下这个优化问题的约束条件。我们首先针对任务要求提出以下三个问题：

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也不能随意选择，需要处于分类间隔的中轴线上。

3）即便取到了合适的方向和截距，并不是所有的训练样本都能成为公式(5.3)里面的支撑向量样本$\boldsymbol{x}_s$。对于一组给定的决策面方程参数$\boldsymbol{\omega}$和γ，该如何找到对应的支持向量$\boldsymbol{x}_s$呢？

尽管上述问题似乎对应三个约束条件，但 SVM 算法可以通过数学技巧将它们融合在一个不等式内。

首先关注第一个问题，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：

$$y_i = \begin{cases} +1 & \text{for 蓝色样本} \\ -1 & \text{for 红色样本} \end{cases} \tag{5.4}$$

如果一个决策面方程能够完全正确地分开训练样本，就会满足下面的公式：

$$\begin{cases} \boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma > 0 \ \ \forall y_i = 1 \\ \boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma < 0 \ \ \forall y_i = -1 \end{cases} \tag{5.5}$$

其次考虑第二个问题，假设决策面正好处于间隔区域的中轴线上，则所有样本到决策面的距离均大于等于d，公式(5.5)就可以进一步写成：

$$\begin{cases} \dfrac{\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma}{\|\boldsymbol{\omega}\|} \geq d, & \forall y_i = 1 \\ \dfrac{\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma}{\|\boldsymbol{\omega}\|} \leq -d, & \forall y_i = -1 \end{cases} \tag{5.6}$$

令两个不等式的左右两边都除 d，可得到：

$$\begin{cases} \boldsymbol{\omega}_d^T \boldsymbol{x}_i + \gamma_d \geq 1, & \forall y_i = 1 \\ \boldsymbol{\omega}_d^T \boldsymbol{x}_i + \gamma_d \leq -1, & \forall y_i = -1 \end{cases} \tag{5.7}$$

其中

$$\boldsymbol{\omega}_d = \frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\| d}, \qquad \gamma_d = \frac{\gamma}{\|\boldsymbol{\omega}\| d} \tag{5.8}$$

把$\boldsymbol{\omega}_d$和γ_d看做是一个决策面方程的参数，很容易可以证明$\boldsymbol{\omega}_d^T \boldsymbol{x}_i + \gamma_d = 0$和$\boldsymbol{\omega}^T \boldsymbol{x} + \gamma = 0$表示同一个决策面。现在，忘记原来的决策面方程参数$\boldsymbol{\omega}$和$\gamma$，将参数$\boldsymbol{\omega}_d$和$\gamma_d$重新命名为$\boldsymbol{\omega}$和$\gamma$。则约束条件转化为：

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

最后考虑第三个问题，处于分类间隔边界上的样本为支持向量，他们对应于点到决策面距离正好为d的情况，对比公式(5.6)和(5.9)，不难理解他们对应于公式(5.9)中等号成立的情况。我们尝试将公式(5.9)给出的约束条件进一步精练，把类别标签y_i和两个不等式左边相乘，形成统一的表述：

$$y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma) \geq 1, \qquad \forall \boldsymbol{x}_i, y_i \tag{5.10}$$

至此，本节开始提出的三个约束在公式(5.9)表达的同一组约束条件中得到了完美的描述。

5.2.4 线性 SVM 优化问题的数学模型

在公式(5.9)中只有支持向量样本能够使得等号成立，即：

$$\begin{cases} \boldsymbol{\omega}^T \boldsymbol{x}_s + \gamma = 1, & \forall y_s = 1 \\ \boldsymbol{\omega}^T \boldsymbol{x}_s + \gamma = -1, & \forall y_s = -1 \end{cases} \tag{5.11}$$

将公式(5.11)带入公式(5.3)，可以得到：

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

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，该问题可以转化为：

$$\boldsymbol{\omega}^*, \gamma^* = \underset{\boldsymbol{\omega},\gamma}{\operatorname{argmax}} 2d = \underset{\boldsymbol{\omega},\gamma}{\operatorname{argmax}} \frac{\mathbf{2}}{\|\boldsymbol{\omega}\|} = \underset{\boldsymbol{\omega},\gamma}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{\omega}\|^2 \tag{5.13}$$

结合前面的分析，一旦决策面的方向确定，分类间隔区域就确定了；而决策面的位置必须位于分类间隔的中轴线上，因此最优决策面的位置也将随之确定。因此真正决定目标函数大小的是决策面方程的方向参数矢量$\boldsymbol{\omega}$，这一推论显然在公式(5.13)中得到了验证。

除了目标函数的变形外，至此可以给出线性 SVM 最优化问题的数学描述如下：

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

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，因此每一个样本对应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 2.3 节介绍的最优化方法，发现这类问题可以使用拉格朗日对偶法进行求解。

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 2.3 节介绍的拉格朗日对偶方法，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：

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

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：

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

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)，则线性 SVM 的拉格朗日对偶问题，可以写为：

$$\max_{\boldsymbol{\alpha}}\left[\sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j\right] \tag{5.24}$$

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

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。相比于公式(5.14)描述的 SVM 原始问题，公式(5.24)描述的对偶问题的自变量从$\boldsymbol{\omega},\gamma$变为$\alpha_i, i = 1,\dots,m$，变量个数从$d+1$变为$m$，看起来似乎更加复杂了。但是约束条件的不等式形式得到了很大的简化，为后续的优化问题求解奠定了基础。

5.3.2 SVM 求解的 KKT 条件

根据 2.3 节介绍的有约束优化问题的求解思路，结合 SVM 的实际情况，则公式(5.14)的最优解的 KKT 条件可以写为：

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

公式(5.25)的第三个式子表示：对于任意训练样本$\boldsymbol{x}_i$，要么$\alpha_i=0$，要么$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i+\gamma)-1=0$。前面分析过，$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i+\gamma)-1=0$意味着样本$\boldsymbol{x}_i$是支撑向量。所以得到这样一个结论：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)，可以看出对偶问题的目标函数仅由$\alpha_i>0$的样本决定，即仅由支撑向量决定。另外需要提到的是，尽管在公式(5.25)中我们只提到了 3 条 KKT 条件，但在公式(5.24)给出的对偶问题中，新的自变量α_i还需要满足$\sum_{i=1}^{m}\alpha_i y_i=0$的约束条件，这一点是可行解的必要条件，也是最优解的必要条件，对于后续的求解具有重要的意义。

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$\alpha_i,i=1,2,\ldots,m$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，是一种离散搜索与局部优化相结合的规划策略。该算法以寻找满足 KKT 条件的解为目标，将多变量搜索问题转化为迭代的单变量优化问题。具体算法过程分为以下 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单变量优化问题；
3) 优化求解：找到满足约束条件的单变量优化问题的最优解
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。

(1) 变量选择

为了将找到满足 KKT 条件的m个变量α_i的多变量搜索问题转化为迭代的单变量优化问题，首先随机初始化m个拉格朗日乘子 $\alpha_i,i=1,2,\ldots,m$；在每次迭代优化之前，选择当前条件下最有利于 SMO 算法寻优过程的两个变量α_j和α_k作为待优化的变量，锁定其他没有选中的拉格朗日乘子$\alpha_i,\forall i\neq j,k$；再通过$\sum_{i=1}^{m}\alpha_i y_i=0$的约束条件下找到两者的函数关系，从而将多变量问题转变为单变量问题。

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。首先在$\alpha_i>0$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：

$$j=\underset{i:\alpha_i>0}{\operatorname{argmax}}(\zeta_i)=\underset{i:\alpha_i>0}{\operatorname{argmax}}(|y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i+\gamma)-1|) \tag{5.26}$$

这里ζ_i是满足$\alpha_i>0$条件的样本$\boldsymbol{x}_i$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：

$$\boldsymbol{\omega} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i \tag{5.27}$$

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:

$$y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma) - 1 = 0, \forall \alpha_i > 0 \tag{5.28}$$

因此在算法没有收敛前，每一个$\boldsymbol{x}_i$带入公式(5.28)都能得到一个γ，因此一般使用公式(5.29)来估计截距γ。

$$\gamma = \frac{1}{|\mathbb{S}|} \sum_{x_i \in \mathbb{V}} (y_i - \boldsymbol{\omega}^T \boldsymbol{x}_i) \tag{5.29}$$

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

其次在$\alpha_i = 0$的非支撑向量样本中逐个验证 KKT 条件$y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma) > 1$，因此$1 - y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma)$的值越大说明该样本破坏 KKT 条件越严重，则乘子α_j的选择可以写为：

$$j = \underset{i:\alpha_i = 0}{\operatorname{argmax}}(\zeta_i) = \underset{i:\alpha_i = 0}{\operatorname{argmax}}\big(1 - y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma)\big) \tag{5.30}$$

对$\alpha_i > 0$和$\alpha_i = 0$两种情况进行综合，则表示 KKT 条件破坏程度的ζ_i可以写为：

$$\zeta_i = \begin{cases} |y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma) - 1|, & \alpha_i > 0 \\ 1 - y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma), & \alpha_i = 0 \end{cases} \tag{5.31}$$

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(5.27)和(5.29)，在$\alpha_i > 0$的范围内寻找公式(5.26)的最优解。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(5.32)：

$$\zeta_i = |y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma) - 1| \leq \epsilon, \forall \alpha_i > 0 \tag{5.32}$$

ϵ是一个事先给定的小正数，是验证α_i是否满足 KKT 条件的阈值，则开始对所有$\alpha_i = 0$的样本验证是否满足$y_i(\boldsymbol{\omega}^T \boldsymbol{x}_i + \gamma) - 1 \geq 0$的 KKT 条件，并根据公式(5.30)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：其基本策略是尽可能选择在优化后具有较大变化的变量作为α_k，以加速优化进程，设某个变量α_i被选中后更新得到最优解α_i^*，则α_k的选择可以写为：

$$k = \underset{i:\alpha_i > 0, i \neq j}{\operatorname{argmax}} |\Delta \alpha_i| = \underset{i:\alpha_i > 0, i \neq j}{\operatorname{argmax}} |\alpha_k^* - \alpha_k| \tag{5.33}$$

关于α_k^*如何求解，以及$\alpha_k^* - \alpha_k$是否可以在变量选择前进行估计的问题，我们将在优化求解部分加以说明。

(2) 问题转化

当待优化变量α_j和α_k被选出后，由于其他变量$\alpha_i, i \neq j,k$被锁定为常数，则公式(5.24)描述的m个变量的最优化问题，将会转化为α_j和α_k的二元优化问题。考虑到$y_i = +1$ 或-1，所以有$y_i^2 = 1, \forall i = 1,2,\dots,m$，则对偶问题的目标函数和约束条件可以写为公式(5.34)：

$$\begin{aligned} &\min_{\alpha_j,\alpha_k} \left[\frac{1}{2}\left(\alpha_j^2 K_{jj} + \alpha_k^2 K_{kk}\right) + \alpha_j\alpha_k y_j y_k K_{jk} + y_j\alpha_j v_j + y_k\alpha_k v_k - \left(\alpha_j + \alpha_k\right)\right] \\ &\text{s.t. } y_j\alpha_j + y_k\alpha_k = -\sum_{i\neq j,k} y_i\alpha_i = C,\ \alpha_j,\alpha_k \geq 0 \end{aligned} \tag{5.34}$$

其中

$$K_{jk} = K\left(\boldsymbol{x}_j, \boldsymbol{x}_k\right) = \boldsymbol{x}_j^T\boldsymbol{x}_k, \qquad j,k = 1,2,\dots,m \tag{5.35}$$

$$v_j = \boldsymbol{x}_j^T \sum_{i\neq j,k} \alpha_i y_i \boldsymbol{x}_i = \sum_{i\neq j,k} \alpha_i y_i K_{ij},\ v_k = \boldsymbol{x}_k^T \sum_{i\neq j,k} \alpha_i y_i \boldsymbol{x}_i = \sum_{i\neq j,k} \alpha_i y_i K_{ik} \tag{5.36}$$

此时，观察公式(5.34)的线性约束条件，同时考虑$y_j^2 = y_k^2 = 1$，则α_j可以写成α_k的表达式：

$$\alpha_j = y_j(C - y_k\alpha_k) \tag{5.37}$$

考虑约束条件$\alpha_j \geq 0$，则有：

$$\begin{aligned} y_j(C - y_k\alpha_k) &\geq 0 \\ y_j y_k \alpha_k &\leq y_j C \end{aligned} \tag{5.38}$$

进而推出两种情况：

$$\alpha_k \begin{cases} \leq y_j C, & \text{if } y_k y_j = 1 \\ \geq y_j C, & \text{if } y_k y_j = -1 \end{cases} \tag{5.39}$$

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$y_k y_j = 1$时：如果$y_j C \geq 0$，则$\alpha_k \in [0, y_j C]$；如果$y_k C < 0$，则α_k的可行解区域为空,则需要重新选取α_k。
- 情况 2：当$y_k y_j = -1$时：$\alpha_k \in \left[\max\left(0, y_j C\right), +\infty\right)$；

在明确了上述取值范围后，将公式(5.37)带入公式(5.34)，则对偶问题的目标函数变为α_k的单变量函数$L(\alpha_k)$：

$$\begin{aligned} L(\alpha_k) = \frac{1}{2}&\left((C - y_k\alpha_k)^2 K_{jj} + \alpha_k^2 K_{kk}\right) + y_k\alpha_k(C - y_k\alpha_k)K_{jk} + (C - y_k\alpha_k)v_j + y_k\alpha_k v_k \\ &- \alpha_k - y_j(C - y_k\alpha_k) \end{aligned} \tag{5.40}$$

用$L(\alpha_k)$对α_k求导，得到：

$$\frac{\partial L(\alpha_k)}{\partial \alpha_k} = (\alpha_k - Cy_k)K_{jj} + \alpha_k K_{kk} + (Cy_k - 2\alpha_k)K_{jk} - y_k v_j + y_k v_k - 1 + y_k y_j \tag{5.41}$$

令该导数为 0，将公式(5.41)中的常数 1 替换为$y_k y_k$，求解α_k^*：

$$\begin{aligned} \alpha_k^*\left(K_{jj} + K_{kk} - 2K_{jk}\right) &= y_k\left(C\left(K_{jj} - K_{jk}\right) + \left(v_j - v_k\right) + \left(y_k - y_j\right)\right) \\ \alpha_k^* &= \frac{y_k\left[C\left(K_{jj} - K_{jk}\right) + \left(v_j - v_k\right) + \left(y_k - y_j\right)\right]}{\left(K_{jj} + K_{kk} - 2K_{jk}\right)} \end{aligned} \tag{5.42}$$

此时，我们结合公式(5.2)，(5.19)对公式(5.36)给出的v_j的定义式进行推导：

$$\begin{aligned} v_j &= \boldsymbol{x}_j^T \sum_{i \neq j,k} \alpha_i y_i \boldsymbol{x}_i \\ &= \boldsymbol{x}_j^T \left(\sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i - \sum_{i=j,k} \alpha_i y_i \boldsymbol{x}_i \right) \\ &= \left(\boldsymbol{x}_j^T \boldsymbol{\omega} + \gamma\right) - \alpha_j y_j K_{jj} - \alpha_k y_k K_{jk} - \gamma \\ &= g\left(\boldsymbol{x}_j\right) - \alpha_j y_j K_{jj} - \alpha_k y_k K_{jk} - \gamma \end{aligned} \tag{5.43}$$

同理可以推出:

$$v_k = g(\boldsymbol{x}_k) - \alpha_j y_j K_{jk} - \alpha_k y_k K_{kk} - \gamma \tag{5.44}$$

将公式(5.43)(5.44)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(5.42)，合并所有的$K_{kk}, K_{jj}, K_{jk}$项，可以得到：

$$\begin{aligned} \alpha_k^* &= \frac{\alpha_k\left(K_{jj} + K_{kk} - 2K_{jk}\right) + y_k\left(\left(g\left(x_j\right) - y_j\right) - \left(g\left(x_k\right) - y_k\right)\right)}{\left(K_{jj} + K_{kk} - 2K_{jk}\right)} \\ \alpha_k^* &= \alpha_k + y_k \frac{e_j - e_k}{\mathcal{Z}_{jk}} \end{aligned} \tag{5.45}$$

其中

$$e_i = g(\boldsymbol{x}_i) - y_i, \qquad i = j, k \tag{5.46}$$

$$\mathcal{Z}_{jk} = K_{jj} + K_{kk} - 2K_{jk} = \boldsymbol{x}_j^T \boldsymbol{x}_j - 2\boldsymbol{x}_j^T \boldsymbol{x}_k + \boldsymbol{x}_k^T \boldsymbol{x}_k = \left\|\boldsymbol{x}_j - \boldsymbol{x}_k\right\|^2 \tag{5.47}$$

因此，$\mathcal{Z}_{jk}$等价于样本$\boldsymbol{x}_j, \boldsymbol{x}_k$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(5.39)下方），如果公式(5.45)计算出的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$\alpha_k^*$最近的取值范围的边界，即$\alpha_k(t+1) = 0$或$\alpha_k(t+1) = y_j C$；这样可以避免$\alpha_k^*$破坏 KKT 条件。此后，再利用公式(5.37)计算出$\alpha_j(t+1)$的数值。

第二个因子α_k的选择方案细化：此时，我们可以为第二个因子α_k的选择重新给出更加具体的计算方案。根据公式(5.33)和(5.45)，为使变化量$|\Delta\alpha_k|$尽可能的大，α_k的最优选择应为：

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

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。但这样做在程序上会比较繁琐。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$\mathcal{Z}_{ji}, i \neq j$作为样本$\boldsymbol{x}_j, \boldsymbol{x}_i$之间的欧氏距离的平方，在优化过程中部发生变化，因此可以事先计算并存储好矩阵$\boldsymbol{\mathcal{Z}} = [\mathcal{Z}_{ji} | i, j = 1, \dots, m]$。$e_i = g(\boldsymbol{x}_i) - y_i, i = 1,2, \dots, m$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，先对所有的$\alpha_i, i \neq j$计算对应的$|\Delta\alpha_i|$，再从大到小进行搜索，当使用当前的α_i带来的目标函数$L(\alpha_i)$取值（见公式(5.40)）的下降幅度ΔL（指相对上一次迭代的目标函数值的降幅）大于给定的阈值$T_{\Delta L}$，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。

(3) 终止条件

对偶问题变量$\alpha_i, i = 1, \dots, m$优化过程的终止条件即要求$\alpha_i, i = 1, \dots, m$的解在精度$\epsilon$内满足 KKT 条件。根据公式(5.25)，可以分为非支撑向量($\forall\alpha_i = 0$)和支撑向量($\forall\alpha_i = 0$)两种情况。

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

其中函数$g(\boldsymbol{x})$的参数$\boldsymbol{\omega}, \gamma$的计算，详见公式(5.27)，(5.29)。

综合上述内容，线性可分 SVM 问题的 SMO 算法步骤可以归纳如下：

♦ **SMO 算法步骤**

输入:	训练数据$X = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \dots, (\boldsymbol{x}_m, y_m) \mid \boldsymbol{x}_i \in \mathbb{R}^d, y_i \in \{+1, -1\}, i = 1,2, \dots, m\}$ 精度ϵ
步骤:	
1	初始化$t = 0$
2	初始化拉格朗日乘子$\boldsymbol{\alpha}(t) = \{0, \dots, 0\}$ 初始化距离矩阵$\boldsymbol{\mathcal{Z}} = [\mathcal{Z}_{ij} \mid i, j = 1, \dots, m]$（公式(5.47)）
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(5.27)，(5.29)）
输出:	参数$\boldsymbol{\omega}, \gamma$

图 5-3 SMO 算法步骤

♦ **例子：基于 SMO 算法的线性可分 SVM 问题求解**

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(5.47)计算矩阵$\boldsymbol{Z}$，如下：

$$\boldsymbol{Z}=\left[\left\|\boldsymbol{x}_i-\boldsymbol{x}_j\right\|^2 | i,j=1,\ldots,m\right]=\begin{bmatrix}0 & 2 & 1\\ 2 & 0 & 1\\ 1 & 1 & 0\end{bmatrix}$$

2) 选取变量

首先，计算 KKT 条件的破坏程度ζ：利用公式(5.27)，(5.29)，求出$\boldsymbol{\omega}=[0,0]^T, \gamma=0$；根据公式(5.2)，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(5.31)得到：

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

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(5.30)，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$\boldsymbol{\Delta\alpha}$选择第二个待优化变量$\boldsymbol{\alpha_k}$。利用公式(5.46)计算所有样本的预测误差：$e_1=1, e_2=1, e_3=-1$，带入公式(5.33)，得到$\Delta\alpha_2=0, \Delta\alpha_3=2$，根据公式(5.48)选择$k(t)=3$。即最终选择$\alpha_1$和$\alpha_3$作为当前轮次的待更新变量。

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(5.39)的要求，先确定$y_j C$的数值：

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

考虑到$y_1 y_3=-1$，根据公式(5.39)下的情况 2，得到当前α_k的取值范围为$[0,+\infty)$。

最后，确定此轮迭代的更新结果$\alpha_j(t+1)$和$\alpha_k(t+1)$。由于$\alpha_3^*=2$在α_k的取值范围$[0,+\infty)$内，令$\alpha_k(t+1)=a_3^*=2$。然后将其带入公式(5.37)，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“●”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。

在实际应用中，SMO 算法比我们在例题中介绍的还要复杂一些，主要是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	a_i	$\boldsymbol{\omega}$	γ	$g(x_i)$	ζ_i	j	$\Delta\alpha_i$	k	a_k^*	y_jC	y_jy_k	$\alpha_k^{(t+1)}$	$\alpha_j^{(t+1)}$
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]	–	–	–	–	–	–	–	–

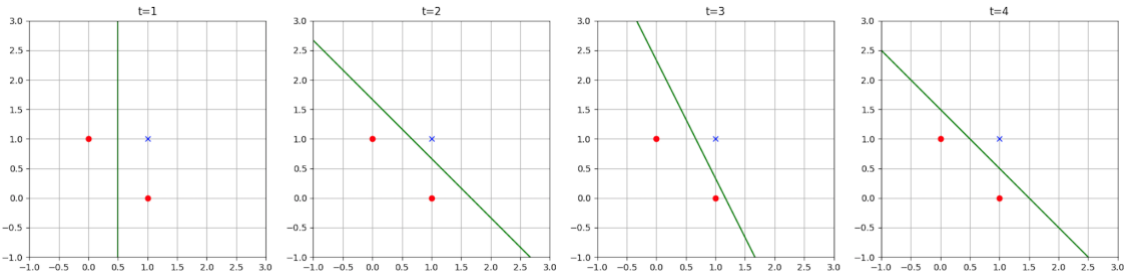


图 5-4　例题的 SMO 算法迭代优化过程中决策面变化情况

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。对于线性不可分问题，如果仍然使用线性分类器，则应允许模型在训练结束后仍然存在一定的错分现象。本节介绍的软间隔线性 SVM 算法就是遵循这一思路解决 SVM 问题的一种方法。

5.4.1 松弛变量的引入

对于一个线性不可分问题，要求线性分类器$g(\boldsymbol{x}) = \boldsymbol{\omega}^T\boldsymbol{x} + \gamma$对所有训练样本均满足约束条件$y_i(\boldsymbol{\omega}^T\boldsymbol{x_i} + \gamma) \geq 1, \forall \boldsymbol{x}_i$（详见公式(5.10)）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，实现对约束条件的宽松化，如下所示：

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

公式(5.50)的意思是样本可以在松弛变量ξ_i对应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，因此我们称具有松弛变量的算法为“软间隔”SVM 算法。

5.4.2 软间隔最大化问题

虽然引入松弛变量可以放宽 SVM 优化问题的约束条件，但从优化的角度看，我们希望松弛变量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 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。因此，软间隔 SVM 的优化问题可以写为：

$$\begin{aligned} &\min_{\boldsymbol{\omega},\gamma,\xi} \left(\frac{1}{2}\|\omega\|^2 + \lambda \sum_{i=1}^{m} \xi_i \right) \\ &\text{s. t.} \quad y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) \geq 1 - \xi_i, \qquad i = 1,2,\dots,m \\ &\qquad\qquad \xi_i \geq 0, \qquad\qquad i = 1,2,\dots,m \end{aligned} \tag{5.51}$$

在目标函数方面，$\lambda > 0$称为惩罚因子，λ越大则模型对于样本侵入软间隔区域的重视程度越高。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λ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

相比于原始的线性 SVM 算法，软间隔 SVM 优化问题额外增加了 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线性约束条件下的凸优化问题，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

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

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：

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

需要注意的是，由于公式(5.57)的约束，每一对α_i和β_i只相当于一个有效变量，因此公式(5.58)中并不包含变量β_i，也就不需要对β_i求极大值，因此最终的对偶问题写为：

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

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：$0 \leq \alpha_i \leq C, i = 1,2,\dots,m$

$$\begin{aligned}\max_{\boldsymbol{\alpha}} & \sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j \\ s.t. \quad & \sum_{i=1}^{m}\alpha_i y_i=0 \\ & 0\le\alpha_i\le\lambda, i=1,2,\dots,m\end{aligned} \tag{5.60}$$

- **软间隔 SVM 问题的 SMO 算法改造**

对比软间隔 SVM 的优化问题与线性可分 SVM 的优化问题(见公式(5.24))，可以发现两者仅在α_i的约束条件上增加了一个上界λ。因此，只需要对线性 SVM 的 SMO 求解算法中两个待优化因子α_j和α_k的搜索范围加以限制即可。根据α_j和α_k之间的关系（见公式(5.37)），同时考虑约束条件$0\le\alpha_k,\alpha_j\le\lambda$，则有：

$$\begin{cases} y_j(C-y_k\alpha_k)\ge 0 \\ \alpha_k\ge 0 \\ y_j(C-y_k\alpha_k)\le\lambda \\ \alpha_k\le\lambda \end{cases} \tag{5.61}$$

当$y_k y_j=1$时：

$$\begin{cases}\alpha_k\le y_j C\\ \alpha_k\ge 0\\ \alpha_k\ge y_j C-\lambda\\ \alpha_k\le\lambda\end{cases}\Rightarrow\begin{cases}\alpha_k\ge L\\ \alpha_k\le H\end{cases}\Rightarrow\begin{cases}L=\max[0,y_j C-\lambda]\\ H=\min[y_j C,\lambda]\end{cases} \tag{5.62}$$

当$y_k y_j=-1$时：

$$\begin{cases}\alpha_k\ge -y_j C\\ \alpha_k\ge 0\\ \alpha_k\le\lambda-y_j C\\ \alpha_k\le\lambda\end{cases}\Rightarrow\begin{cases}\alpha_k\ge L\\ \alpha_k\le H\end{cases}\Rightarrow\begin{cases}L=\max[0,-y_j C]\\ H=\min[\lambda-y_j C,\lambda]\end{cases} \tag{5.63}$$

公式(5.62)和(5.63)意味着在$y_k y_j=1$和$y_k y_j=-1$两种情况下，α_k各有对应的下界L和上界H。因此软间隔 SVM 问题的 SMO 算法只需要根据$y_k y_j$的取值，结合公式(5.45)计算得到的最优解α_k^*更新$\alpha_k(t+1)$即可，如公式(5.64)所示。其他优化求解过程均与线性 SVM 问题的 SMO 求解方法一致。

$$\alpha_k(t+1)=\begin{cases}L, & \alpha_k^*<L\\ \alpha_k^*, & L\le\alpha_k^*\le H\\ H, & \alpha_k^*>H\end{cases} \tag{5.64}$$

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

(5.27)和(5.28))，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）。在标准的线性 SVM 模型假设下，上述支撑作用在几何层面上很好理解，即支撑向量样本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。为了深入理解软间隔 SVM 模型中样本$\boldsymbol{x}_i$与最终间隔区域在几何层面的关系，需要先了解软间隔 SVM 问题的 KKT 条件，具体如公式(5.63)所示：

$$\begin{cases} \alpha_i \geq 0 & (1) \\ y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i \geq 0 & (2) \\ \alpha_i(y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i) = 0 & (3) \\ \beta_i \geq 0 & (4), \quad i = 1,2,\dots,m \\ \xi_i \geq 0 & (5) \\ \beta_i\xi_i = 0 & (6) \\ C - \alpha_i - \beta_i = 0 & (7) \end{cases} \tag{5.65}$$

当$\alpha_i = 0$时，根据 KKT 条件(7)：$C - \alpha_i - \beta_i = 0$，有$\beta_i = C$；再根据 KKT 条件(6)：$\beta_i\xi_i = 0$，推出$\xi_i = 0$；再根据 KKT 条件在$\alpha_i = 0$时样本$\boldsymbol{x}_i$是非支撑向量，所以$y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1 + \xi_i > 0$，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**。

当$\alpha_i > 0$时，根据 KKT 条件(3)，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有影响，对应于 **Case2**。

在 Case2 中，α_i仍有两种可能性。第一种情况：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**。第二种情况：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。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**。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**。表 5-2 对上述几种情况进行了总结，图 5-5 则给出了上述情况下样本对几何分布示例。

表 5-2　软间隔 SVM 中样本所处位置分析

<table>
<tr><td rowspan="3">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="2">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:
$\alpha_i = C$,</td><td>Case 2-2-1：
$\xi_i < 1$,</td></tr>
</table>

		$\beta_i = 0,$ $\xi_i > 0$	$\boldsymbol{x}_i$在间隔区域内正确的一侧； $\boldsymbol{x}_i$能被正确分类。
			Case 2-2-2： $\xi_i \geq 1,$ $\boldsymbol{x}_i$越过了决策面或在决策面上； $\boldsymbol{x}_i$被错误分类。

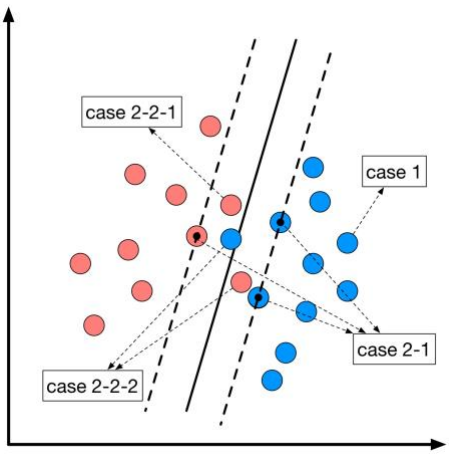


图 5-5　软间隔 SVM 中不同α_i和ξ_i对应对样本几何分布情况

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，根据公式(5.65)给出的 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：

$$d(\boldsymbol{x}_i) = \frac{y_i(\boldsymbol{\omega}^T\boldsymbol{x}_i + \gamma) - 1}{\|\boldsymbol{\omega}\|} = \frac{\xi_i}{\|\boldsymbol{\omega}\|} \tag{5.66}$$

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λ来调和两种相互矛盾的目标。

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，我们还可以从公式(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。首先对优化问题目标函数进行适当的适当变形如下：

$$\min_{\omega,\gamma} \sum_{i=1}^{m} \ell_h(\boldsymbol{x}_i) + \mu\|\omega\|^2 \tag{5.67}$$

其中

$$\ell_h(\boldsymbol{x}) = \begin{cases} 1 - y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma), & y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) < 1 \\ 0, & y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) \geq 1 \end{cases} \tag{5.68}$$

$$\mu = \frac{1}{2\lambda} \tag{5.69}$$

显然，只需要在公式(5.67)的目标函数上乘以系数λ就可以得到公式(5.51)所描述的软间隔 SVM 的目标函数，所以在变形前后的两个最优化问题有着相同的最优解。我们称函数$\ell_h(\boldsymbol{x})$为**合叶损失函数(Hinge loss function)**，如果将$y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma)$视为自变量，则$\ell_h(\boldsymbol{x})$的函数曲线如图 5-6 所示。

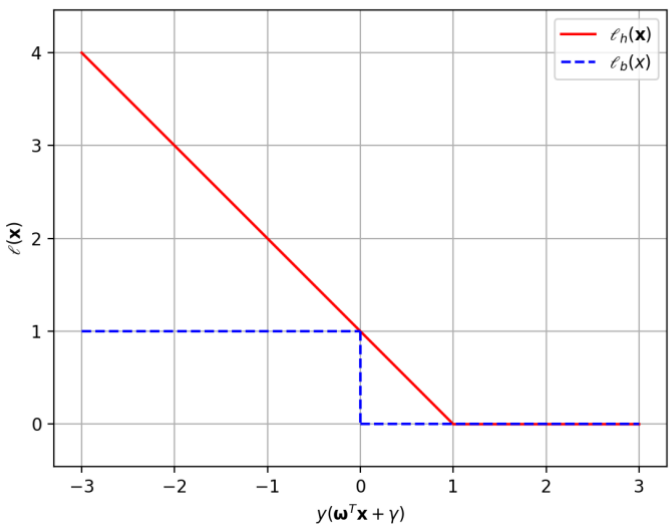


图 5-6　合叶损失函数$\ell_h(\boldsymbol{x})$与 0-1 阶跃损失函数$\ell_b(\boldsymbol{x})$曲线图

从中可以看出合叶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$y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) > 1$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$y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) \leq 1$时，样本$\boldsymbol{x}$开始侵入间隔区域，甚至可能出现$y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) \leq 0$的错误分类情况，因此会产生分类代价$1 - y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma)$。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：

$$\ell_b(\boldsymbol{x}) = \begin{cases} 1, & y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) \leq 0 \\ 0, & y(\boldsymbol{\omega}^T\boldsymbol{x} + \gamma) > 0 \end{cases} \tag{5.70}$$

事实上感知器算法就采用了$\ell_b(\boldsymbol{x})$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，没有考虑样本到决策面的远近情况，因此模型泛化能力不够好。而合叶损失函数$\ell_h(\boldsymbol{x})$不但处处连续可导，且在样本进入间隔区域后函数值会随着样本到决策面的距离发生变化，有助于增强模型的泛化能力。实际上除了合叶损失函数以外，还有其他相似形状的函数也可以作为评价模型分类错误情况的损失函数，例如指数损失函数$\ell_{\exp}(\boldsymbol{x})=\exp\big(-y(\boldsymbol{\omega}^T\boldsymbol{x}+\gamma)\big)$或对数损失函数$\ell_{\log}(\boldsymbol{x})=\log\big[1+\exp\big(-y(\boldsymbol{\omega}^T\boldsymbol{x}+\gamma)\big)\big]$等。它们统称为**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目标，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相似，即可以找到无穷多个解使得回归误差为零。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。

综上所述，合叶损失函数$\ell_h(\boldsymbol{x})$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，减轻过拟合风险，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，但同时又相互矛盾。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。

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$g(\boldsymbol{x})$设定为非线性函数，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两方面的挑战：

1) 判别函数$g(\boldsymbol{x})$的非线性函数形式与参数设定问题。判别函数应该使用多项式、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) 非线性目标函数的优化求解问题。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。这意味着之前学习的 KKT 条件和拉格朗日对偶等技巧都会失效，我们是否能够找到求解有约束非线性优化问题的有效策略？

尽管存在上述挑战，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却又非常巧妙的思路——**非线性映射+核技巧。**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。根据第二章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，那么该数据集在$\mathcal{X}$的任意线性变换或$\mathcal{X}$的任意线性子空间中仍然是不可分的。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加以形象的说明。

如图 5-7 (a)所示，红色“●”和蓝色“＋”号分别对应于二维空间$\mathcal{X} = x_1 \times x_2$中的两类，$x_1, x_2$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$x_3 = x_1^2 + x_2^2$，并将两类样本映射到新构造的三维空间$\Phi = x_1 \times x_2 \times x_3$中，其分布如图 5-7 (b)所示。显然训练数据在原二维空间$\mathcal{X}$中是线性不可分的，但经过简单的非线性映射，在新的三维空间Φ中就变成了一个线性可分问题，只需要构造一个$x_3 = 1$的决策平面，就可以完美解决原始数据的分类问题。

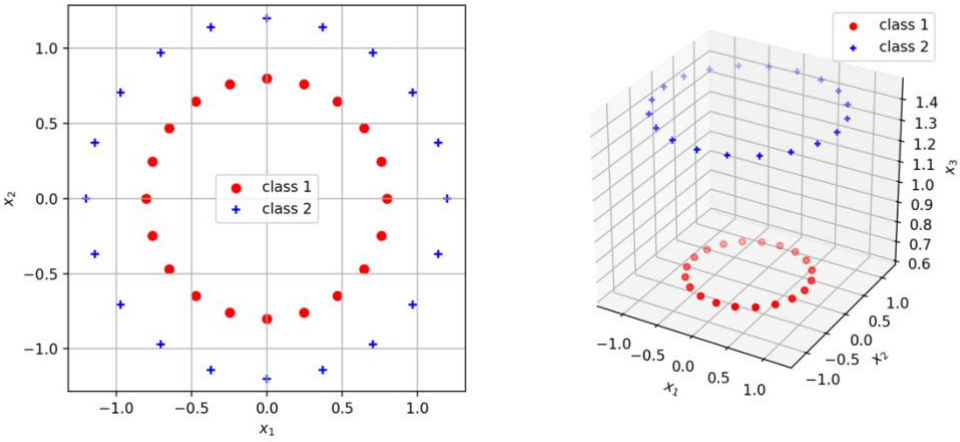


图 5-7　数据非线性映射举例示意图

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：

$$\phi(\boldsymbol{x}):\ \mathcal{X} \to \Phi \tag{5.71}$$

借助非线性映射函数$\phi(\boldsymbol{x})$，线性不可分问题似乎已经被解决了。但如何找到一个非线性映射函数$\phi(\boldsymbol{x})$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“聪明”的解决思路。

5.5.2 核化 SVM

先不去考虑非线性映射函数$\phi(\boldsymbol{x})$的具体数学形式，仅从$\phi(\boldsymbol{x})$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)，可以写出对应的优化问题如下：

$$\begin{aligned} &\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ &s.t. \quad \sum_{i=1}^{m} \alpha_i y_i = 0 \\ &\qquad 0 \le \alpha_i \le C, i = 1,2,\dots,m \end{aligned} \tag{5.72}$$

其中

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j), \forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \Omega(\mathbf{x}) \tag{5.73}$$

函数$K(x_i, x_j)$称为核函数，$\phi(\boldsymbol{x})$是对应的映射函数。观察公式(5.73)可以发现，核函数$K(x_i, x_j)$的输出是两个向量$\phi(\boldsymbol{x}_i)$和$\phi(\boldsymbol{x}_j)$的内积，是一个标量。从形式上看，对于一个给定的映射$\phi(\boldsymbol{x})$，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$K(x_i, x_j)$，却可能存在多个甚至无穷多个映射$\phi(\boldsymbol{x})$能够与之对应，而$\phi(\boldsymbol{x})$对应的希尔伯特空间Φ的维度也存在多种可能性，理论上，空间Φ的维度甚至可以是无穷大。

回想面向线性 SVM 问题的 SMO 算法，公式(5.35)中的K_{jk}可以看成是的核函数在映射函数$\phi(\boldsymbol{x}) = \boldsymbol{x}$条件下的特例，这意味着在 SMO 算法的求解过程中，其实并不需要计算非线性映射$\phi(\boldsymbol{x})$的具体数值，只需要能够计算$K(x_i, x_j), \forall i, j$即可。此外，即使在 SVM 模型的应用阶段，我们也并不需要具体计算出Φ空间的中线性决策面$\boldsymbol{\omega}$参数，这是因为$\phi(\boldsymbol{x})$总是与模型参数$\boldsymbol{\omega}$同时以$\boldsymbol{\omega}^T\phi(\boldsymbol{x})$的形式出现的。参照公式(5.55)，用非线性映射后的$\phi(\boldsymbol{x})$取代原特征空间$\mathcal{X}$中的样本$\boldsymbol{x}$，则有

$$\boldsymbol{\omega}^T\phi(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i y_i \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) \tag{5.74}$$

这意味着非线性映射$\phi(\boldsymbol{x})$在整个 SVM 问题的求解和应用过程中从未单独出现过，而总是通核函数K的形式出现，因此我们并不需要知道$\phi(\boldsymbol{x})$的具体数学形式，只需要能够对任意样本$\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{X}$，计算出核函数值$K(\boldsymbol{x}_i, \boldsymbol{x}_j)$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(5.73)，核函数$K(\boldsymbol{x}_i,\boldsymbol{x}_j)$必须能够写成非线性映射$\boldsymbol{\phi}(\boldsymbol{x}_i)$和$\boldsymbol{\phi}(\boldsymbol{x}_j)$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$K$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数学形式的问题。

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

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

2） 映射函数$\boldsymbol{\phi}(\boldsymbol{x})$能够将原向量$\boldsymbol{x}$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$\mathcal{X}\times\mathcal{X}$直积空间上的对称函数，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。

函数K的 Gram 矩阵$\mathcal{K}$定义为：

$$\mathcal{K}=\begin{bmatrix}K(\boldsymbol{x}_1,\boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_1,\boldsymbol{x}_m)\\ \vdots & \ddots & \vdots\\ K(\boldsymbol{x}_m,\boldsymbol{x}_1) & \cdots & K(\boldsymbol{x}_m,\boldsymbol{x}_m)\end{bmatrix} \tag{5.75}$$

关于定理 5.1 的证明应分别从核函数的定义即$K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\boldsymbol{\phi}(\boldsymbol{x}_i)^T\boldsymbol{\phi}(\boldsymbol{x}_j)$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$\boldsymbol{\phi}(\boldsymbol{x})\in\mathcal{H}$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$\boldsymbol{\Phi}$被称为**再生希尔伯特空间**。由于上述证明过程涉及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[1]。

尽管有了定理 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对于大多数本书的读者可能仍然是困难的。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）

$$K(\boldsymbol{x}_i,\boldsymbol{x}_j)=(\boldsymbol{x}_i^T\boldsymbol{x}_j+1)^q, q\geq 1 \tag{5.76}$$

根据公式(5.2)和(5.74)，判别函数$g(\boldsymbol{x})$可以写为：

$$g(\boldsymbol{x})=\boldsymbol{\omega}^T\boldsymbol{x}+\gamma=\sum_i^m a_i y_i(\boldsymbol{x}_i^T\boldsymbol{x}+1)^q+\gamma \tag{5.77}$$

其中$\alpha_i, i=1,2,\dots,m$为 SVM 对偶问题最优解，γ为对应的决策面方程常数项。

[1] 具体内容建议查阅李航老师的《统计学习方法》第二版

2） 高斯核函数（Gaussian kernnel function）

$$K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\exp\left(-\frac{\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2}{2\sigma^2}\right),\sigma^2>0 \tag{5.78}$$

对应的判别函数为：

$$g(\boldsymbol{x})=\sum_i^m a_i y_i \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}_i\|^2}{2\sigma^2}\right)+\gamma \tag{5.79}$$

3） Sigmoid 核函数（Sigmoidal kernel function）

$$K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\tanh\left(\beta\boldsymbol{x}_i^T\boldsymbol{x}_j+\theta\right),\beta,\theta>0 \tag{5.80}$$

对应的判别函数为：

$$g(\boldsymbol{x})=\sum_i^m a_i y_i \tanh\left(\beta\boldsymbol{x}_i^T\boldsymbol{x}+\theta\right)+\gamma \tag{5.81}$$

5.5.4 理解核化 SVM

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。这种影响包含“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。线性 SVM 算法可以看做是核函数$K(\boldsymbol{x}_i,\boldsymbol{x}_j)=\boldsymbol{x}_i^T\boldsymbol{x}_j$的核化 SVM 算法的特例，此时支撑样本$\boldsymbol{x}_i$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$\boldsymbol{x}_i$对决策面的贡献与基本度量$\boldsymbol{x}_i^T\boldsymbol{x}$值呈多项式关系。这意味着$\boldsymbol{x}_i^T\boldsymbol{x}$值越大，支撑样本$\boldsymbol{x}_i$的相对作用越大，且这种趋势随着参数$q$的增加成指数增加。因此，我们可以得到这样的结论：**“支撑样本$\boldsymbol{x}_s$对于决策面$g(x)=0$的贡献比例服从基本度量的某种非线性变化，具体表现为支撑样本$\boldsymbol{x}_s$与测试样本$\boldsymbol{x}$的相似性越高，则$\boldsymbol{x}_s$对于最终决策的贡献比例就越大。”**

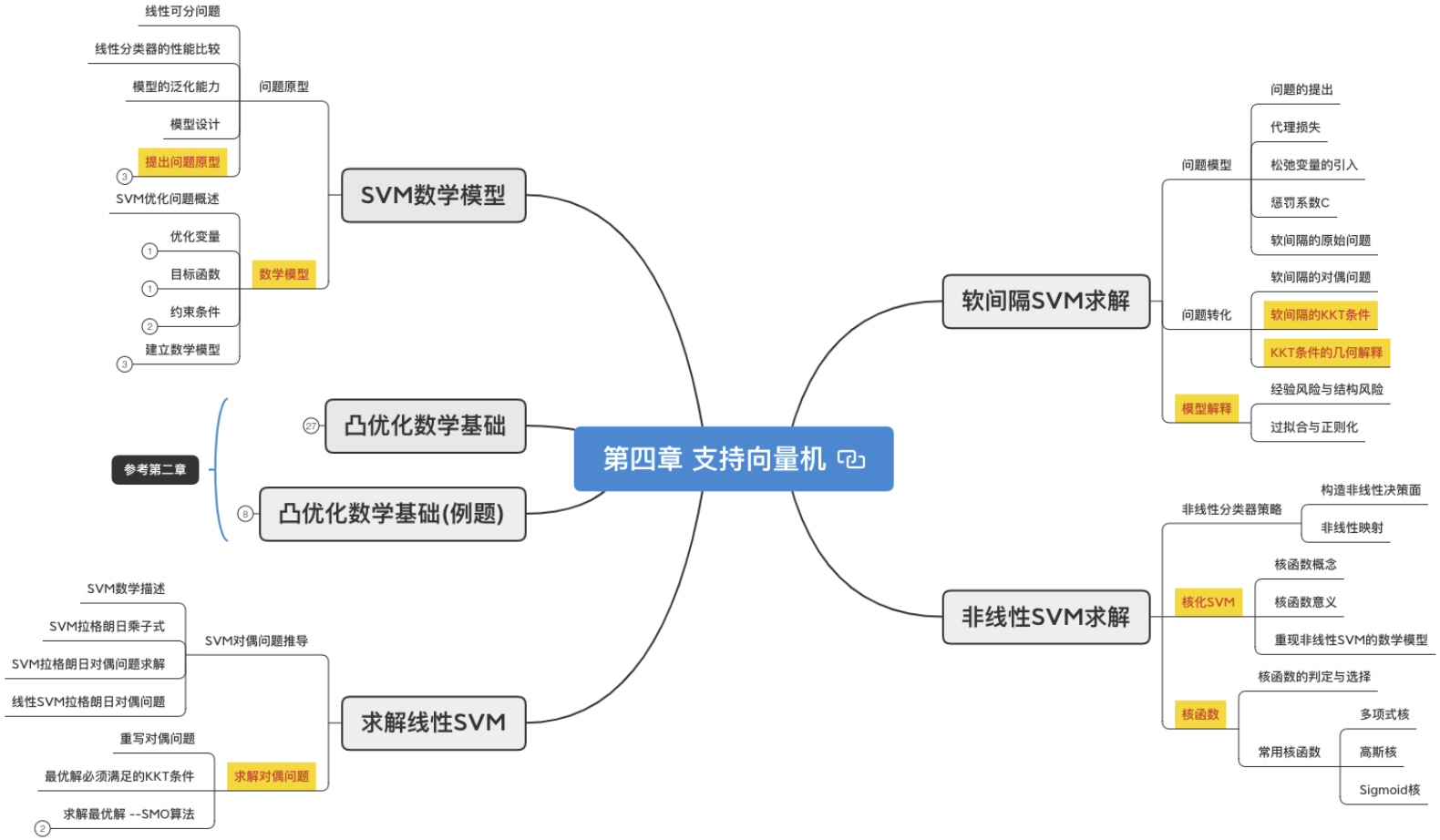
从另一个角度看，核化 SVM 类似于基于某种非线性距离度量的 K 近邻分类器。仍以多项式核函数为例，判别函数$g(\boldsymbol{x})$可以写为标签$y_i\in\{1,-1\}$某种线性组合：

$$g(\boldsymbol{x})=\sum_i^m \eta_i y_i+\gamma,\qquad \eta_i=a_i\left(\boldsymbol{x}_i^T\boldsymbol{x}+1\right)^q \tag{5.82}$$

显然，η_i反应了支撑向量$\boldsymbol{x}_i$参与测试样本$\boldsymbol{x}$分类决策的权重。当代表非线性程度的q较大时，不同支撑向量的权重差异被拉大，$g(\boldsymbol{x})$的符号将主要由与测试样本$\boldsymbol{x}$最相似的几个支撑向量决定；在极端情况下，当q趋近于无穷大时，$g(\boldsymbol{x})$的符号由$\boldsymbol{x}$的最近邻样本决定，此时的核化 SVM 相当于一个最近邻分类器。因此，原则上看，只要q值设定的足够大，核化 SVM 一定可以将所有训练样本都正确分类，因为每个训练样本的最近邻样本都是其自身。

尽管“非线性映射+核技巧”的核化SVM方案在理论上可以将任意低维空间中的线性不可分问题转化为高维空间中的线性可分问题，但由于核函数的函数形式和参数是手动选择的，在实际应用中并不一定能确保所有的训练样本都被正确分类。从另一个角度看，即便可以做到这一点，我们仍然希望适当降低核函数的非线性程度，从而提高模型的泛化能力，降低过拟合风险。因此，在大多数情况下，我们会将核化SVM算法与软间隔SVM算法相结合，通过设定合适的比例参数λ来调节经验风险与结构风险的平衡，以期获得更好的泛化性能。

本章思维导图


第四章 支持向量机
SVM数学模型
问题原型
线性可分问题
线性分类器的性能比较
模型的泛化能力
模型设计
提出问题原型
数学模型
SVM优化问题概述
优化变量
目标函数
约束条件
建立数学模型
凸优化数学基础
凸优化数学基础(例题)
参考第二章
求解线性SVM
SVM对偶问题推导
SVM数学描述
SVM拉格朗日乘子式
SVM拉格朗日对偶问题求解
线性SVM拉格朗日对偶问题
求解对偶问题
重写对偶问题
最优解必须满足的KKT条件
求解最优解 --SMO算法
软间隔SVM求解
问题模型
问题的提出
代理损失
松弛变量的引入
惩罚系数C
软间隔的原始问题
问题转化
软间隔的对偶问题
软间隔的KKT条件
KKT条件的几何解释
模型解释
经验风险与结构风险
过拟合与正则化
非线性SVM求解
非线性分类器策略
构造非线性决策面
非线性映射
核化SVM
核函数概念
核函数意义
重现非线性SVM的数学模型
核函数
核函数的判定与选择
常用核函数
多项式核
高斯核
Sigmoid核

本章习题