第六章 SVM 算法

By Xian2207, 13689903575, wszhangxian@126.com

6.1 间隔与支持向量

假设样本空间为$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$，$y \in \{-1, +1\}$，分类学习基本思想就是在训练集 D 中找到一个划分超平面，将不同类别的样本分开。

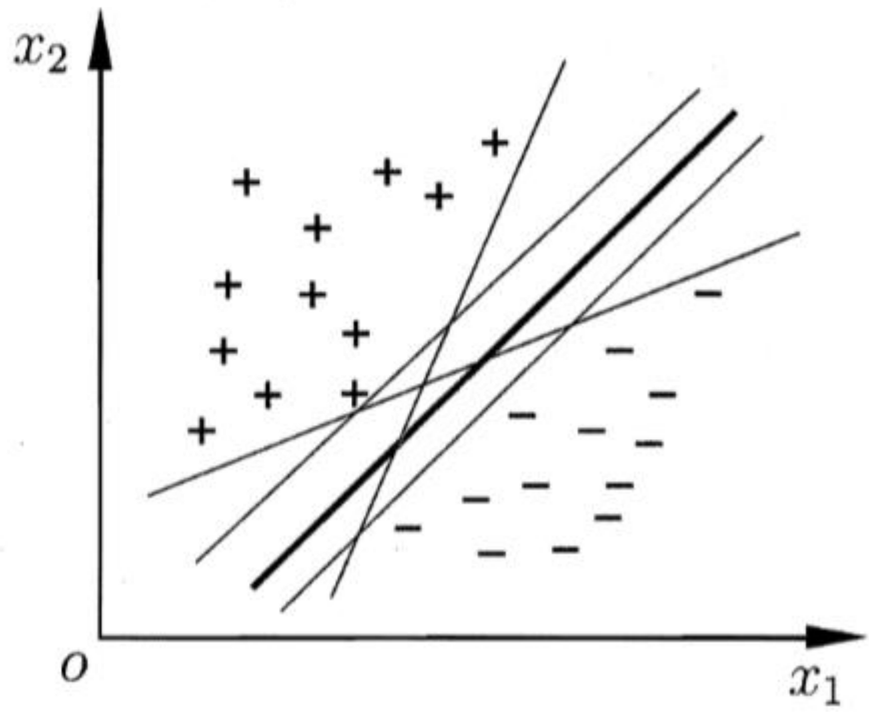

图 6.1 存在多个划分超平面将两类训练样本分开

划分超平面可用如下方程描述

$$f(x) = w^T x + b \tag{6.1}$$

w为法向量，决定了超平面的方向；b为位移项，决定了超平面与原点之间的距离，故超平面可被法向量和位移确定。样本空间任一点x到超平面(w, b)的距离可表示为

$$r = \frac{w^T x + b}{||w||} \tag{6.2}$$

假设超平面(w, b)可将 D 中样本分开，即$(x_i, y_i) \in D$，若$y_i = +1$，则$w^T x_i + b > 0$；$y_i = -1$，则$w^T x_i + b < 0$. 令

$$\begin{cases} w^T x_i + b \geq 1, & y_i = +1 \\ w^T x_i + b \leq -1, & y_i = -1 \end{cases} \tag{6.3}$$

如下图所示

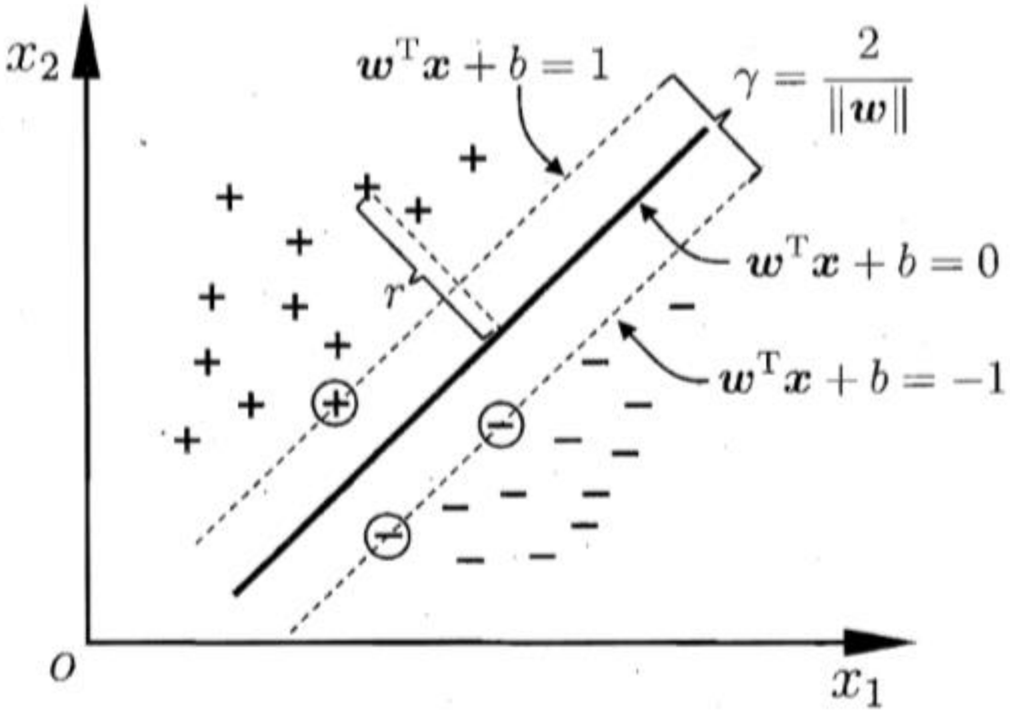

图 6.2　支持向量与间隔

距离超平面（实线）最近的几个训练样本点被称为"支持向量"（support vector），这些点可使 6.3 式成立。两个异类超平面（虚线）到达超平面的距离之和为"间隔"（margin），如下表示

$$r=\frac{2}{||w||} \qquad 6.4$$

欲找到"最大间隔"（maximum margin）划分超平面，就要找合适的w和b，使r最大，即

$$max\frac{2}{||w||}s.t.\,y_i(w^Tx_i+b)\geq 1, i\in(1,2,\dots,m) \qquad 6.5$$

以上求最大，相当于倒数求最小，即

$$min\frac{1}{2}||w||^2\ s.t.\ y_i(w^Tx_i+b)\geq 1, i\in(1,2,\dots,m) \qquad 6.6$$

以上即支持向量机 Support Vector Machine-SVM 的基本型。

6.2 对偶问题

对式 6.6 添加拉格朗日乘子$\alpha_i\geq 0$，可得其对偶问题（dual problem），则该问题的拉格朗日函数可写为

$$L(w,b,\alpha)=\frac{1}{2}||w||^2+\sum_{i=1}^{m}\alpha_i[1-y_i(w^Tx_i+b)] \qquad 6.8$$

其中$\alpha=(\alpha_1;\ \alpha_2;\dots;\ \alpha_m)$，求关于 w 和 b 的偏导数，使其为 0，可得

$$w=\sum_{i=1}^{m}(\alpha_i y_i x_i) \qquad 6.9$$

$$0=\sum_{i=1}^{m}(\alpha_i y_i) \qquad 6.10$$

将 6.9 代入 6.8，再将$L(w,b,\alpha)$中的 w 和 b 消去，再考虑 6.10 约束，就得到 6.6 的对偶问题

$$\max_{\alpha}=\sum_{i=1}^{m}\alpha_i-\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j x_i^T x_j$$

$$\text{s.t.}\ \sum_{i=1}^{m}\alpha_i y_i=0\,,\alpha_i\geq 0,i=1,2,\ldots,m \qquad 6.11$$

解出α后，求出w和b即可得到模型

$$f(x)=w^T x+b=\sum_{i=1}^{m}\alpha_i y_i x_i^T x+b \qquad 6.12$$

从对偶问题解出α即拉格朗日乘子，恰好对应训练样本(x_i,y_i)，注意到 6.6 式有不等式约束，因此上述过程需满足（Karush-Kuhn-Tucker）条件，即

$$\begin{gathered}\alpha_i\geq 0;\\ y_i f(x_i)-1\geq 0;\\ \alpha_i(y_i f(x_i)-1)\geq 0;\end{gathered} \qquad 6.13$$

对任意样本，总会有$\alpha_i=0$或$y_i f(x_i)=1$。若α_i为 0，则 6.12 式变为$f(x)=b$， 不会对$f(x)$产生任何影响。如果$\alpha_i>0$，则必有$y_i f(x_i)=1$，所对应的样本点恰好在最大间隔边界上。故支持向量机有个重要性质：训练完成后，大部分训练样本不必保留，最终模型仅与支持向量有关。求解α开销正比于样本数，为高效求解，SMO（Sequential Minimal Optimization）算法应用而生。算法思路是先固定α_i之外的其他变量，然后利用这些变量导出α_i。步骤为，先选择一对需要更新的α_i,α_j两个参数，固定其他参数，求解 6.11，更新这对参数。然后利用这对参数更新上一步固定的参数，再重复前面步骤。注意到只需选取α_i和α_j中某一个不满足 KTT 条件（6.13 式），目标函数就会在迭代后减小。一般情况下，SMO 县选取违背 KKT 条件程度最大的参数，第二个参数应选择是目标函数减小最快的变量（开销较大），因此 SMO 选取的两变量所对应样本之间间隔最大。

SMO 具体做法是：仅优化两个参数，重写 6.11 的约束

$$\alpha_i y_i+\alpha_j y_j=c,\text{where}\ \alpha_i\geq 0,\alpha_j\geq 0,c=-\sum_{k\neq i,j}\alpha_k y_k \qquad 6.14$$

利用此式和 6.11，消去α_j即可解出α_i，进而求解α_j。至于偏移项b

$$b=\frac{1}{|S|}\sum_{s\in S}(y_s-\sum_{i\in S}\alpha_i y_i x_i^T x) \qquad 6.15$$

其中$S=\{i\mid\alpha_i>0,i=1,2,\ldots,m\}$。

6.3 核函数

上述章节建立在两类线性可分，但现实任务中，样本空间多存在异或问题，即难以找到一个

能正确划分两类样本的超平面，如下图

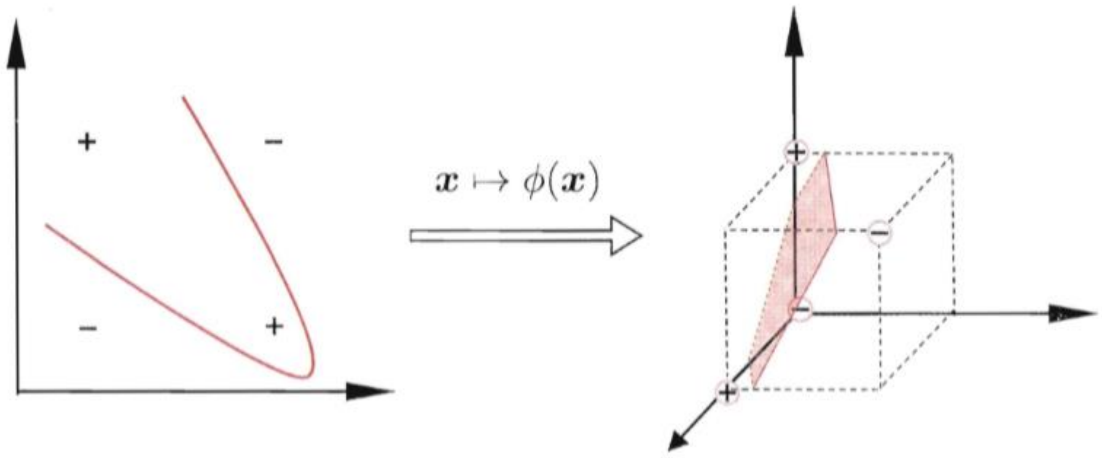

图 6.3 异或问题与非线性映射

对此问题，可将图 6.3 左图从二维映射到高维特征空间，从而找到合适的划分超平面（红色不规则四边形）。如果原始空间是有限维的，即属性数有限，那么一定存在高维特征空间使样本可分。若令$\phi(x)$表示x映射后的特征向量，于是在特征空间中划分超平面所对应的模型表示为

$$f(x) = w^T\phi(x) + b \qquad 6.19$$

类似

$$min\frac{1}{2}||w||^2 \; s.t. \; y_i(w^T\phi(x) + b) \geq 1, i \in (1,2,\ldots,m) \qquad 6.20$$

其对偶问题为

$$\max_{\alpha} = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \phi(x_i)^T\phi(x_j)$$

$$\text{s.t.} \sum_{i=1}^{m}\alpha_i y_i = 0, \alpha_i \geq 0, i = 1,2,\ldots,m \qquad 6.21$$

由于特征空间维数可能很高，甚至可能是无穷维，直接计算$\phi(x_i)^T\phi(x_j)$很困难，为此设想以下函数

$$k(x_i, x_j) = \phi(x_i)^T\phi(x_j) \qquad 6.22$$

代入 6.21 可得

$$\max_{\alpha} = \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j k(x_i, x_j)$$

$$\text{s.t.} \sum_{i=1}^{m}\alpha_i y_i = 0, \alpha_i \geq 0, i = 1,2,\ldots,m \qquad 6.23$$

求解后可得

$$f(x) = w^T\phi(x) + b \qquad 6.24$$

$$= \sum_{i=1}^{m} \alpha_i y_i \phi(x_i)^T \phi(x_j) + b$$

$$= \sum_{i=1}^{m} \alpha_i y_i k(x_i, x_j) + b$$

这里的$k(x_i, x_j)$即为“核函数”（kernel function）。该式表明模型最优化解可通过训练样本的核函数展开，这个展开式叫“支持向量展示”（support vector expansion）。一般情况下，如果知道$\phi(x)$具体形式，则可写出核函数。若在未知$\phi(x)$的具体形式下，要判断该函数是否是有核函数，关键看核矩阵（kernel matrix）是否是半正定的，即特征根乘积大于等于 0.

$$\mathbf{K} = \begin{bmatrix} \kappa(\boldsymbol{x}_1, \boldsymbol{x}_1) & \cdots & \kappa(\boldsymbol{x}_1, \boldsymbol{x}_j) & \cdots & \kappa(\boldsymbol{x}_1, \boldsymbol{x}_m) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \kappa(\boldsymbol{x}_i, \boldsymbol{x}_1) & \cdots & \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) & \cdots & \kappa(\boldsymbol{x}_i, \boldsymbol{x}_m) \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \kappa(\boldsymbol{x}_m, \boldsymbol{x}_1) & \cdots & \kappa(\boldsymbol{x}_m, \boldsymbol{x}_j) & \cdots & \kappa(\boldsymbol{x}_m, \boldsymbol{x}_m) \end{bmatrix}$$

对于任意一个半正定核矩阵，总能找到与其对应的映射$\phi(x)$形式。也就是说，任何一个核函数都隐式定义了一个称为“再生希尔伯特空间”（reproducing kernel Hilbert space, RKHS）的特征空间。既然我们希望样本在特征空间线性可分，故上述特征空间的好坏将决定 SVM 的性能。一般我们不知道什么样的核函数是合适的，它只是隐式定义。所以，核函数选择称为 SVM 的最大变数，如果选择不合适，意味着样本映射到了一个不合适的高维特征空间，导致性能不佳。常见核函数为

表 6.1 常用核函数

名称	表达式	参数
线性核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j$	
多项式核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j)^d$	$d \geqslant 1$ 为多项式的次数
高斯核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\lVert\boldsymbol{x}_i - \boldsymbol{x}_j\rVert^2}{2\sigma^2}\right)$	$\sigma > 0$ 为高斯核的带宽(width)
拉普拉斯核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\lVert\boldsymbol{x}_i - \boldsymbol{x}_j\rVert}{\sigma}\right)$	$\sigma > 0$
Sigmoid 核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \tanh(\beta \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j + \theta)$	tanh 为双曲正切函数，$\beta > 0$, $\theta < 0$

还可通过函数组合得到核函数，例如假设k_1, k_2为核函数，则对于任一正数γ_1, γ_2的线性组合也是核函数

$$\gamma_1 k_1 + \gamma_2 k_2 \qquad 6.25$$

$$k_1 \otimes k_2(x, z) = k_1(x, z) k_2(x, z) \qquad 6.26$$

$$k(x, z) = g(x) k_1(x, z) g(z) \qquad 6.27$$

6.4 软间隔与正则化

现实任务中往往很难确定合适的核函数使样本空间线性可分。即便某个核函数使训练集在特城空间中线性可分，如何判断结果是否过拟合？于是“软间隔”被提出，即允许 SVM 在一些样本上出错，所有样本不必必须划分正确，不必满足约束 6.3 式（硬间隔：样本必须划分正确且满足 6.3 式子）。如图所示

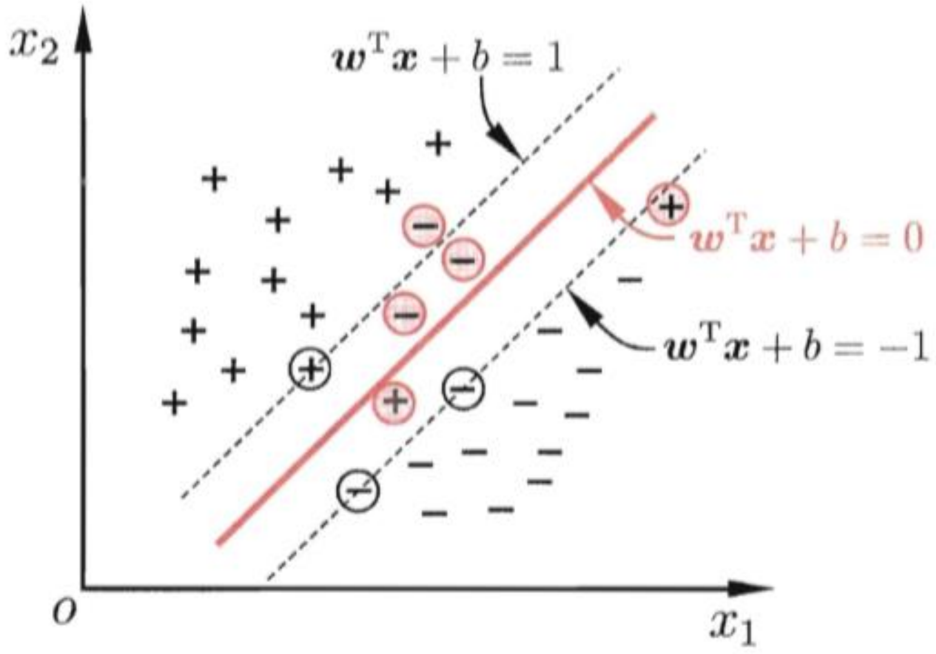

图 6.4 软间隔示意图. 红色圈出了一些不满足约束的样本.

软间隔可允许不满足如下约束

$$\{y_i(w^Tx_i+b)\geq 1 \qquad 6.28$$

最大化间隔时，不满足约束的样本应尽可能少，最优化目标变为

$$\min_{w,b}\frac{1}{2}||w||^2+C\sum_{\ell=1}^{m}\ell_{0/1}[y_i(w^Tx_i+b)-1] \qquad 6.29$$

其中$C>0$是一个常数，$\ell_{0/1}$是“0/1 损失函数”

$$\ell_{0/1}(z)=\begin{cases}1, & \text{if } z<0\\ 0, & \text{otherwise}\end{cases} \qquad 6.30$$

若 C 无穷大，则 6.29 迫使所有样本满足 6.28，于是 6.29 等价于 6.6,；当 C 取有限值，式 6.29 允许样本不满足约束。然而，$\ell_{0/1}$一般非凸，非连续，数学性质不好，使 6.29 不易求解。于是替代损失函数（surrogate loss）用来代替$\ell_{0/1}$。

$$\text{hinge 损失: } \ell_{hinge}(z)=\max(0,1-z)\ ; \qquad (6.31)$$

$$\text{指数损失(exponential loss): } \ell_{exp}(z)=\exp(-z)\ ; \qquad (6.32)$$

$$\text{对率损失(logistic loss): } \ell_{log}(z)=\log(1+\exp(-z))\ . \qquad (6.33)$$

若采用 hinge 损失，则 6.29 变为

$$\min_{\boldsymbol{w},b} \ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}\max\left(0, 1 - y_i\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b\right)\right) \ . \tag{6.34}$$

引入“松弛变量”（slack variable）$\xi_i \geq 0$，6.34 变为

$$\min_{\boldsymbol{w},b,\xi_i} \ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}\xi_i \tag{6.35}$$

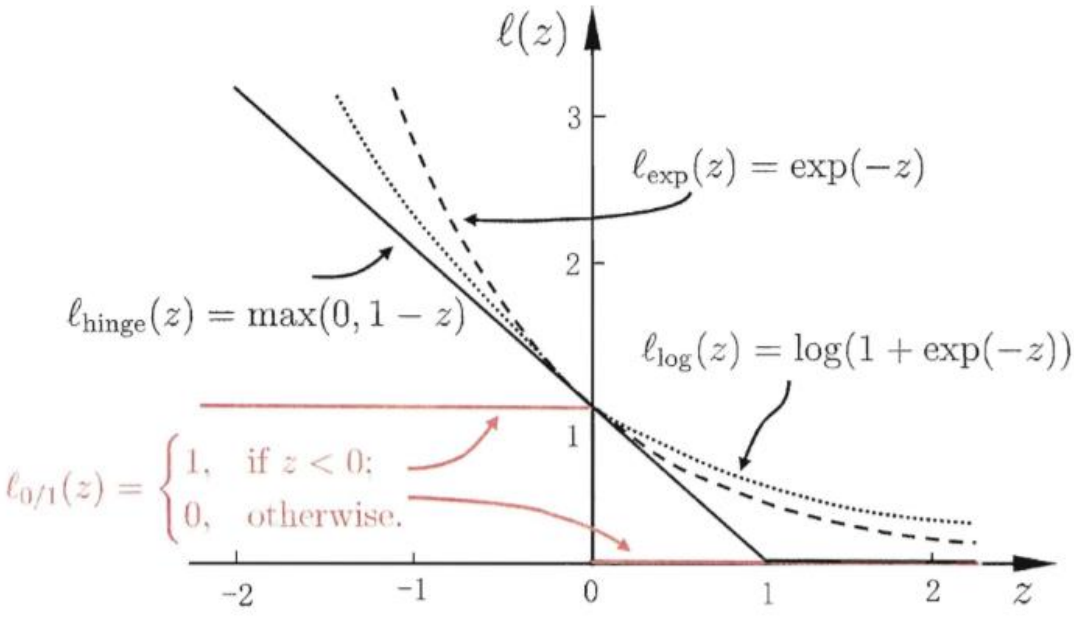

图 6.5 三种常见的替代损失函数：hinge损失、指数损失、对率损失

$$\text{s.t.} \ \ y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) \geqslant 1 - \xi_i$$

$$\xi_i \geqslant 0 \ , \ i = 1, 2, \ldots, m.$$

以上为常见的“软间隔支持向量机”。通过拉格朗日乘子方法可得

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}, \boldsymbol{\xi}, \boldsymbol{\mu}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}\xi_i + \sum_{i=1}^{m}\alpha_i\left(1 - \xi_i - y_i\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b\right)\right) - \sum_{i=1}^{m}\mu_i\xi_i \ , \tag{6.36}$$

其中$\alpha_i \geq 0, \mu_i \geq 0$，令$L(w, b, \alpha, \xi, \mu)$对$w, b, \xi_i$求偏导

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i \ , \tag{6.37}$$

$$0 = \sum_{i=1}^{m} \alpha_i y_i \ , \tag{6.38}$$

$$C = \alpha_i + \mu_i \ . \tag{6.39}$$

将式 6.37-6.39 代入 6.36 可得对偶问题

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j \\ \text{s.t.} \quad & \sum_{i=1}^{m} \alpha_i y_i = 0 \ , \\ & 0 \leqslant \alpha_i \leqslant C \ , \quad i = 1, 2, \ldots, m \ . \end{aligned} \tag{6.40}$$

对比 6.40 软间隔和 6.11（硬间隔），两者唯一差别就是对偶变量的约束不同，前者$0 \le \alpha_i \le C$，后者是$0 \le \alpha_i$。采用 6.2 节当中同样的算法可解 6.40,。类似于 6.13，软间隔 SVM 的 KKT 条件为

$$\begin{cases} \alpha_i \geqslant 0 \ , \quad \mu_i \geqslant 0 \ , \\ y_i f(\boldsymbol{x}_i) - 1 + \xi_i \geqslant 0 \ , \\ \alpha_i \left(y_i f\left(\boldsymbol{x}_i\right) - 1 + \xi_i \right) = 0 \ , \\ \xi_i \geqslant 0 \ , \ \mu_i \xi_i = 0 \ . \end{cases} \tag{6.41}$$

于是, 对任意训练样本 $(\boldsymbol{x}_i, y_i)$, 总有 $\alpha_i = 0$ 或 $y_i f(\boldsymbol{x}_i) = 1 - \xi_i$. 若 $\alpha_i = 0$, 则该样本不会对 $f(\boldsymbol{x})$ 有任何影响; 若 $\alpha_i > 0$, 则必有 $y_i f(\boldsymbol{x}_i) = 1 - \xi_i$, 即该样本是支持向量: 由式(6.39) 可知, 若 $\alpha_i < C$, 则 $\mu_i > 0$, 进而有 $\xi_i = 0$, 即该样本恰在最大间隔边界上; 若 $\alpha_i = C$, 则有 $\mu_i = 0$, 此时若 $\xi_i \leqslant 1$ 则该样本落在最大间隔内部, 若 $\xi_i > 1$ 则该样本被错误分类. 由此可看出, 软间隔支持向量机的最终模型仅与支持向量有关, 即通过采用 hinge 损失函数仍保持了稀疏性.

可以发现，如果使用对率损失函数 ℓ_{log} 来替代式(6.29)中的 0/1 损失函数，则几乎就得到了对率回归模型(3.27)．实际上，支持向量机与对率回归的优化目标相近，通常情形下它们的性能也相当．对率回归的优势主要在于其输出具有自然的概率意义，即在给出预测标记的同时也给出了概率，而支持向量机的输出不具有概率意义，欲得到概率输出需进行特殊处理 [Platt, 2000]；此外，对率回归能直接用于多分类任务，支持向量机为此则需进行推广[Hsu and Lin, 2002]．另一方面，从图 6.5 可看出，hinge 损失有一块“平坦”的零区域，这使得支持向量机的解具有稀疏性，而对率损失是光滑的单调递减函数，不能导出类似支持向量的概念，因此对率回归的解依赖于更多的训练样本，其预测开销更大．

我们还可以把式(6.29)中的 0/1 损失函数换成别的替代损失函数以得到其他学习模型，这些模型的性质与所用的替代函数直接相关，但它们具有一个共性：优化目标中的第一项用来描述划分超平面的“间隔”大小，另一项 $\sum_{i=1}^{m}\ell(f(\boldsymbol{x}_i),y_i)$ 用来表述训练集上的误差，可写为更一般的形式

$$\min_{f} \quad \Omega(f)+C\sum_{i=1}^{m}\ell(f(\boldsymbol{x}_i),y_i), \tag{6.42}$$

其中 $\Omega(f)$ 称为“结构风险”(structural risk)，用于描述模型 f 的某些性质；第二项 $\sum_{i=1}^{m}\ell(f(\boldsymbol{x}_i),y_i)$ 称为“经验风险”(empirical risk)，用于描述模型与训练数据的契合程度；C 用于对二者进行折中．从经验风险最小化的角度来看，$\Omega(f)$ 表述了我们希望获得具有何种性质的模型(例如希望获得复杂度较小的模型)，这为引入领域知识和用户意图提供了途径；另一方面，该信息有助于削减假设空间，从而降低了最小化训练误差的过拟合风险．从这个角度来说，式(6.42)称为“正则化”(regularization)问题，$\Omega(f)$ 称为正则化项，C 则称为正则化常数．L_p 范数 (norm) 是常用的正则化项，其中 L_2 范数 $\|\boldsymbol{w}\|_2$ 倾向于 $\boldsymbol{w}$ 的分量取值尽量均衡，即非零分量个数尽量稠密，而 L_0 范数 $\|\boldsymbol{w}\|_0$ 和 L_1 范数 $\|\boldsymbol{w}\|_1$ 则倾向于 $\boldsymbol{w}$ 的分量尽量稀疏，即非零分量个数尽量少．

6.5 支持向量回归

传统回归模型得到的预测值与真实值之间的误差来计算损失，只有当预测值和真实值完全一样，损失才为零。而支持向量回归（support vector regression, SVR）则允许我们容忍预测值与真实值存在误差 e，即仅当预测与真实值差别绝对值大于 e 时才计算损失。如图 6.6 所示，以 f(x)为中心，构建一个宽度为 2e 的间隔带，若训练样本在此间隔带，则被认为预测正确。

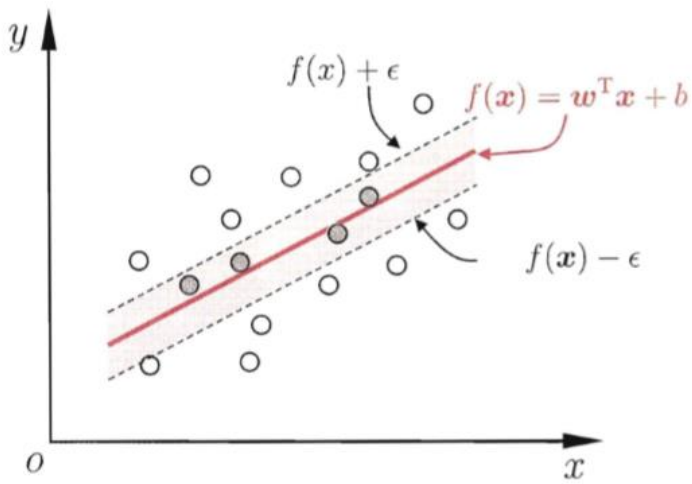

图 6.6 支持向量回归示意图. 红色显示出 ϵ-间隔带, 落入其中的样本不计算损失

于是，SVR 问题转化为

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}\ell_\epsilon(f(\boldsymbol{x}_i)-y_i) \ , \tag{6.43}$$

其中 C 为正则化常数，ℓ_e是 e-不敏感损失（e-insensitive loss）函数

$$\ell_e(z)=\begin{cases}1, & \text{if } |z|\le e\\ |z|-e, & \text{otherwise}\end{cases} \qquad 6.34$$

引入松弛变量$\xi_i,\hat{\xi}_i$，6.43 可重写为

$$\min_{\boldsymbol{w},b,\xi_i,\hat{\xi}_i} \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}(\xi_i+\hat{\xi}_i) \tag{6.45}$$

$$\begin{aligned}\text{s.t. } & f(\boldsymbol{x}_i)-y_i \leqslant \epsilon+\xi_i \ ,\\ & y_i-f(\boldsymbol{x}_i) \leqslant \epsilon+\hat{\xi}_i \ ,\\ & \xi_i \geqslant 0,\ \hat{\xi}_i \geqslant 0\ ,\ i=1,2,\ldots,m.\end{aligned}$$

类似的，引入拉格朗日乘子，$\mu_i \ge 0, \hat{\mu}_i \ge 0, \alpha_i \ge 0, \hat{\alpha}_i \ge 0$

$$
\begin{aligned}
&L(\boldsymbol{w}, b, \boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}, \boldsymbol{\xi}, \hat{\boldsymbol{\xi}}, \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}) \\
&= \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}(\xi_i + \hat{\xi}_i) - \sum_{i=1}^{m}\mu_i\xi_i - \sum_{i=1}^{m}\hat{\mu}_i\hat{\xi}_i \\
&+ \sum_{i=1}^{m}\alpha_i\big(f(\boldsymbol{x}_i) - y_i - \epsilon - \xi_i\big) + \sum_{i=1}^{m}\hat{\alpha}_i\big(y_i - f(\boldsymbol{x}_i) - \epsilon - \hat{\xi}_i\big) . \qquad (6.46)
\end{aligned}
$$

将 6.7 代入 6.46，再令 L 关于$w, b, \xi_i, \hat{\xi}_i$的偏导数为零，可得

$$
\boldsymbol{w} = \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i)\boldsymbol{x}_i \ , \qquad (6.47)
$$

$$
0 = \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i) \ , \qquad (6.48)
$$

$$
C = \alpha_i + \mu_i \ , \qquad (6.49)
$$

$$
C = \hat{\alpha}_i + \hat{\mu}_i \ . \qquad (6.50)
$$

将 6.47-6.50 代入 6.46，可得 SVR 的对偶问题

$$
\begin{aligned}
\max_{\boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}} \quad & \sum_{i=1}^{m} y_i(\hat{\alpha}_i - \alpha_i) - \epsilon(\hat{\alpha}_i + \alpha_i) \qquad (6.51) \\
& - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\hat{\alpha}_i - \alpha_i)(\hat{\alpha}_j - \alpha_j)\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j \\
\text{s.t.} \quad & \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i) = 0 \ , \\
& 0 \leqslant \alpha_i, \hat{\alpha}_i \leqslant C \ .
\end{aligned}
$$

上述过程满足 KKT 条件，即

$$\begin{cases} \alpha_i(f(\boldsymbol{x}_i) - y_i - \epsilon - \xi_i) = 0 \ , \\ \hat{\alpha}_i(y_i - f(\boldsymbol{x}_i) - \epsilon - \hat{\xi}_i) = 0 \ , \\ \alpha_i\hat{\alpha}_i = 0 \ , \ \xi_i\hat{\xi}_i = 0 \ , \\ (C - \alpha_i)\xi_i = 0 \ , \ (C - \hat{\alpha}_i)\hat{\xi}_i = 0 \ . \end{cases} \tag{6.52}$$

可以看出，当且仅当$f(x_i) - y_i - e - \xi_i = 0$时，$\alpha_i$能取非零值。当且仅当$y_i - f(x_i) - e - \hat{\xi}_i = 0$时$\hat{\alpha}_i$能取非零值。换言之，仅当样本不落入 e-间隔带，相应的α_i和$\hat{\alpha}_i$才能取非零值。此外，约束两者等于 0 的情况不能同时成立，故α_i和$\hat{\alpha}_i$中至少有一个为 0. 将 647 代入 6.7， SVR 解的形式为

$$f(\boldsymbol{x}) = \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i)\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x} + b \ . \tag{6.53}$$

能使 6.53 中的$\hat{\alpha}_i - \alpha_i \neq 0$的样本即为 SVR 的支持向量，它们必落在 e-间隔带之外。显然 SVR 支持向量仅是训练样本的一部分，其解具有稀疏性。

由 KKT 条件(6.52)可看出, 对每个样本 $(\boldsymbol{x}_i, y_i)$ 都有 $(C - \alpha_i)\xi_i = 0$ 且 $\alpha_i(f(\boldsymbol{x}_i) - y_i - \epsilon - \xi_i) = 0$. 于是, 在得到 α_i 后, 若 $0 < \alpha_i < C$, 则必有 $\xi_i = 0$, 进而有

$$b = y_i + \epsilon - \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i)\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x} \ . \tag{6.54}$$

因此, 在求解式(6.51)得到 α_i 后, 理论上来说, 可任意选取满足 $0 < \alpha_i < C$ 的样本通过式(6.54)求得 b. 实践中常采用一种更鲁棒的办法: 选取多个(或所有)满足条件 $0 < \alpha_i < C$ 的样本求解 b 后取平均值.

若考虑特征映射形式(6.19), 则相应的, 式(6.47)将形如

$$\boldsymbol{w} = \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i)\phi(\boldsymbol{x}_i) \ . \tag{6.55}$$

将式(6.55)代入(6.19), 则 SVR 可表示为

$$f(\boldsymbol{x}) = \sum_{i=1}^{m} (\hat{\alpha}_i - \alpha_i) \kappa(\boldsymbol{x}, \boldsymbol{x}_i) + b \,, \tag{6.56}$$

其中 $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^{\mathrm{T}} \phi(\boldsymbol{x}_j)$ 为核函数.

6.6 核方法

若不考虑偏置 b，则无论 SVM 还是 SVR，模型总能表示成核函数$k(x, x_i)$的线性组合

$$\min_{h \in \mathbb{H}} \; F(h) = \Omega(\|h\|_{\mathbb{H}}) + \ell(h(\boldsymbol{x}_1), h(\boldsymbol{x}_2), \ldots, h(\boldsymbol{x}_m)) \tag{6.57}$$

其解可为

$$h^*(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i \kappa(\boldsymbol{x}, \boldsymbol{x}_i) \,. \tag{6.58}$$

最优解$h^*(x)$都可表示为核函数的线性组合，基于一系列核函数的学习方法，统称为“核方法”（kernel methods）。最常见的是通过“核化”即引入核函数来将线性学习器拓展为非线性学习器，从而得到“核线性判别分析”（Kernelized Linear Discriminant Analysis，简称 KLDA）。

假设通过某种映射$\phi: \chi \to F$将样本映射到一个特征空间 F，然后在 F 中执行线性判别分析，以求得$h(x) = w^T \phi(x)$，KLDA 的学习目标是

$$\max_{\boldsymbol{w}} J(\boldsymbol{w}) = \frac{\boldsymbol{w}^{\mathrm{T}} \mathbf{S}_b^{\phi} \boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}} \mathbf{S}_w^{\phi} \boldsymbol{w}} \,, \tag{6.60}$$

其中S_b^{ϕ}, S_w^{ϕ}分别为训练样本在特征空间 F 中的类间三都矩阵和类内散度矩阵。令X_i表示第$i \in \{0,1\}$类样本的集合，其样本数为m_i；总样本数$m = m_0 + m_1$. 第i类样本在特征空间 F 中的均值为

$$\boldsymbol{\mu}_i^{\phi} = \frac{1}{m_i} \sum_{\boldsymbol{x} \in X_i} \phi(\boldsymbol{x}) \,, \tag{6.61}$$

其中两个散度矩阵为

$$\mathbf{S}_b^{\phi} = (\boldsymbol{\mu}_1^{\phi} - \boldsymbol{\mu}_0^{\phi})(\boldsymbol{\mu}_1^{\phi} - \boldsymbol{\mu}_0^{\phi})^{\mathrm{T}} \,; \tag{6.62}$$

$$\mathbf{S}_w^{\phi} = \sum_{i=0}^{1} \sum_{\boldsymbol{x} \in X_i} (\phi(\boldsymbol{x}) - \boldsymbol{\mu}_i^{\phi})(\phi(\boldsymbol{x}) - \boldsymbol{\mu}_i^{\phi})^{\mathrm{T}} \,. \tag{6.63}$$

通常很难知道ϕ的具体形式，因此使用核函数$k(x, x_i) = \phi(x_i)^T \phi(x_i)$来隐式表达映射和特征空间 F。记 J(w)为 6.57 中的损失函数，再令$\Omega \equiv 0$，函数$h(x)$表示为

$$h(\boldsymbol{x}) = \sum_{i=1}^{m} \alpha_i \kappa(\boldsymbol{x}, \boldsymbol{x}_i) \ , \tag{6.64}$$

根据 6.59 可得

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i \phi(\boldsymbol{x}_i) \ . \tag{6.65}$$

令 $\mathbf{K} \in \mathbb{R}^{m \times m}$ 为核函数 κ 所对应的核矩阵, $(\mathbf{K})_{ij} = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$. 令 $\mathbf{1}_i \in \{1,0\}^{m \times 1}$ 为第 i 类样本的指示向量, 即 $\mathbf{1}_i$ 的第 j 个分量为 1 当且仅当 $\boldsymbol{x}_j \in X_i$, 否则 $\mathbf{1}_i$ 的第 j 个分量为 0. 再令

$$\hat{\boldsymbol{\mu}}_0 = \frac{1}{m_0}\mathbf{K}\mathbf{1}_0 \ , \tag{6.66}$$

$$\hat{\boldsymbol{\mu}}_1 = \frac{1}{m_1}\mathbf{K}\mathbf{1}_1 \ , \tag{6.67}$$

$$\mathbf{M} = (\hat{\boldsymbol{\mu}}_0 - \hat{\boldsymbol{\mu}}_1)(\hat{\boldsymbol{\mu}}_0 - \hat{\boldsymbol{\mu}}_1)^{\mathrm{T}} \ , \tag{6.68}$$

$$\mathbf{N} = \mathbf{K}\mathbf{K}^{\mathrm{T}} - \sum_{i=0}^{1} m_i \hat{\boldsymbol{\mu}}_i \hat{\boldsymbol{\mu}}_i^{\mathrm{T}} \ . \tag{6.69}$$

于是, 式(6.60)等价为

$$\max_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha}) = \frac{\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{M}\boldsymbol{\alpha}}{\boldsymbol{\alpha}^{\mathrm{T}}\mathbf{N}\boldsymbol{\alpha}} \ . \tag{6.70}$$

显然, 使用线性判别分析求解方法即可得到 $\boldsymbol{\alpha}$, 进而可由式(6.64)得到投影函数 $h(\boldsymbol{x})$.

6.7 实例

参考链接：

https://blog.csdn.net/Chenyukuai6625/article/details/73863594?utm_source=blogxgwz8

注意：上述例子仅有拉格朗日乘子法的数据实例，其他计算很少，一般都是调库。如果非要自己算，建议去 github 搜索 SVM 代码，自己逐行破解即可。