支持向量机

1.线性间隔

分类学习最基本的想法就是基于训练集 D 在样本空间中找到一个超平面，将不同的类别可以划分出来。

在样本空间中划分超平面可通过如下线性方程来求解

$$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}+b=0 \tag{1}$$

样本空间中任意点 $\boldsymbol{x}$

到超平面$(\boldsymbol{x},b)$ 的距离为

$$r=\frac{|\boldsymbol{w}^{\boldsymbol{T}}+b|}{||w||} \tag{2}$$

假设超平面 $(\boldsymbol{w},b)$ 能将训练样本正确分类，即对于 $(\boldsymbol{x}_i,y_i)\in D$,若 $y_i=+1$,则有$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}+b>0$; 若$y_i=-1$, 则有 $\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}+b<0$,令

$$\begin{cases}\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}+b\geq 0, y_i=+1\\ \boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}+b\leq 0, y_i=-1\end{cases} \tag{3}$$

距离超平面最近的几个训练样本点使得公式(3) 的等号成立，他们被称之为支持向量 $(SupportVector)$,两个异类到超平面的距离为

$$\gamma=\frac{2}{\|\boldsymbol{w}\|} \tag{4}$$

想要找到最大间隔的超平面，也就是要找公式(3) 中的约束条件 $\boldsymbol{w}$ 和 b,使得 γ 最大，即

$$\begin{aligned}&\max_{\boldsymbol{w},b}\frac{2}{\|\boldsymbol{w}\|}\\ &s.t.\ y_i(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i}+b)\geq 1,\quad i=1,2,3,\ldots,m.\end{aligned} \tag{5}$$

为了最大化间隔，仅需最大化 $\|\boldsymbol{w}\|^{-1}$ ，这等价于最小化$\|\boldsymbol{w}\|^2$ 于是，公式(5) 可重写为

$$\begin{aligned}&\min_{\boldsymbol{w},b}\frac{1}{2}\|\boldsymbol{w}\|\\ &s.t.\ y_i(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i}+b\geq 1),\ i=1,2,3,\ldots,m\end{aligned} \tag{6}$$

2.对偶问题

求解公式(6)可以得到大间隔超平面模型，

$$f(x) = \boldsymbol{w^T} x + b \tag{7}$$

其中$\boldsymbol{w}$ 和 b 是模型参数，公式(6)是一个凸二次规划 ($Convex\ quadratic\ programming$)问题，可通过计算包进行求解，但是也有更高效的方法。

对公式(6) 使用拉格朗日乘子法可得到其对偶问题($dual\ problem$).具体来说，对每一个约束添加拉格朗日乘子 $a_i \geq 0$,则该问题的拉格朗日函数可写为

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\| + \sum_{i=1}^{m} \alpha(1 - y_i(\boldsymbol{w^T x_i} + b)), \tag{8}$$

其中$\boldsymbol{\alpha} = (\alpha_1; \alpha_2; \ldots; \alpha_n)$.令$L(\boldsymbol{w}, b, \boldsymbol{\alpha})$ 对$\boldsymbol{w}$ 和 b 求偏导为零，可得

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x_i} \tag{9}$$

$$0 = \sum_{i=1}^{m} a_i y_i \tag{10}$$

将公式(9)代入(8) ,即可将$L(\boldsymbol{w}, b, \boldsymbol{\alpha})$ 中的$\boldsymbol{w}$ 和 b 消去，在考虑公式(10)的约束，就得到

$$\begin{aligned} \max_{a} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} a_i a_j y_i y_j \boldsymbol{x_i^T x_j} \\ s.t. \sum_{i=1}^{m} a_i y_i = 0, \\ a_i \geq 0,\ i = 1, 2, \ldots, m \end{aligned} \tag{11}$$

解出$\boldsymbol{\alpha}$ 后，求出w和b 可得到模型

$$\begin{aligned} f(x) &= \boldsymbol{w^T x} + b \\ &= \sum_{i=1}^{m} a_i y_i \boldsymbol{x_i^T} + b \end{aligned} \tag{12}$$

从对偶问题解出的 α_i 是 (8)式中的拉格朗日乘子，对应着训练样本$(\boldsymbol{x}_i, y_i)$.在(6)式中有不等式约束，因此在上述过程中需要满足 KKT 约束。约束公式如下：

$$\begin{cases} a_i \geq 0 \\ y_i f(x_i) - 1 \geq 0 \\ \alpha_i (y_i f(x_i) - 1) \geq 0 \end{cases} \tag{13}$$

在这样的情况下，对任意的训练样本$(\boldsymbol{x}_i, y_i)$,总有一个 $\alpha_i = 0$ 或$y_i f(x_i) = 1$.若$\alpha_i = 0$,则该样本将不会在(12)中出现，也就不会对$f(x)$有任何影响;若$a_i > 0$,则必有$y_i f(x_i) = 1$,所对应的样本点位于最大间隔的边界上，是一个支持向量。这显示出支持向量的一个重要性质：训练完成后，大部分的训练样本都不需要保留，最终模型仅与支持向量有关。

求解公式(11)我们可以使用二次规划算法来求解；但是在实际任务中可能会有很大的开销。因此为了避免额外的开销出现了很多优化算法。其中著名的代表式 SMO 算法。

SMO算法的基本思路是先固定α_i 之外的所有参数，然后求α_i 上的极值。由于 存在约束$\sum_{i=1}^{m} a_i y_i = 0$,若先固定α_i 之外的其他变量，则α_i 变量可由其他变量导出。于是SMO 每次选择两个变量α_i 和 α_j ,并固定其他参数。这样在参数初始化后，SMO 不断执行如下两个步骤直至收敛

- 选取一对需更新的变量α_i 和 α_j
- 固定α_i 和 α_j 之外的参数，求解公式(11)获得更新后的 α_i 和 α_j

SMO 算法之所以高效，由于在固定其他参数后，仅优化两个参数的过程能做到非常高效

3.核函数

在前面的讨论都是平面线性可分的情况，当样本空间不存在一个能正确划分的超平面时。可将原始空间映射到一个更高维的特征空间，使得样本在这个特征空间内线性可分。

令$\phi(\boldsymbol{x})$ 表示将$\boldsymbol{x}$ 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为：

$$f(\boldsymbol{x}) = \boldsymbol{w}^T \phi(\boldsymbol{x}) + b \tag{14}$$

其中$\boldsymbol{x}$ 和 b 是模型参数.类似公式(6)

$$\begin{aligned} &\min_{\boldsymbol{w},b} \frac{1}{2} ||\boldsymbol{w}||^2 \\ &s.t.\, y_i(\boldsymbol{w}^T \phi(\boldsymbol{x}_i) + b) \geq 1, i = 1, 2, \ldots, m \end{aligned} \tag{15}$$

其对偶问题是

$$\begin{aligned} \max_{\boldsymbol{a}} \sum_{i=1}^{m} \alpha_i = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) \\ s.t. \sum_{i=1}^{m} \alpha_i y_i = 0, \\ a_i \geq 0, i = 1, 2, \ldots, m \end{aligned} \tag{16}$$

求解公式(16)涉及到计算 $\phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j)$,这是样本$\boldsymbol{x}_i$与$\boldsymbol{x}_j$ 映射到特征空间之后的内积.由于特征空间维数可能会很高，甚至是无穷维，因此直接计算$\phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j)$ 通常是困难的.为了规避这个障碍.我们可以有如下技巧.

$$k(\boldsymbol{x_i}, \boldsymbol{x_j}) =< \phi(\boldsymbol{x}_i, \phi(\boldsymbol{x}_j) >= \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}_j) \tag{17}$$

即$\boldsymbol{x}_i$ 与$\boldsymbol{x}_j$ 在特征空间的内积等于它们在原始样本空间中通过函数$k(\cdot, \cdot)$ 计算的结果.有这个函数就可以不必直接去计算高维甚至是无穷维特征空间中的内积，于是公式(15)可重写为

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j k(\boldsymbol{x_i}, \boldsymbol{x_j}) \\ s.t. \sum_{i=1}^{m} \alpha_i y_i = 0 \alpha_i \geq 0, i = 1, 2, \ldots, m \end{aligned} \tag{18}$$

求解后得到

$$
\begin{aligned}
f(x) &= \boldsymbol{w}^T \phi(\boldsymbol{x}) + b \\
&= \sum_{i=1}^{m} \alpha_i y_i \phi(\boldsymbol{x}_i)^T \phi(\boldsymbol{x}) + b \\
&= \sum_{i=1}^{m} \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i) + b
\end{aligned} \tag{19}
$$

形如 $k(\cdot,\cdot)$ 的函数就是核函数。公式(19) 显示出模型最优解可通过训练样本的核函数展开，这一展开式亦称"支持向量展式"

在实际应用中,如果我们已知合适映射$\phi(\cdot)$ 的具体形式则可写出核函数，但是在实际应用中，可能不确定具体的 $\phi(\cdot)$ 是什么形式。有如下定理

定理 （核函数） 令 χ 为输入空间，$k(\cdot,\cdot)$ 是定义在$\chi * \chi$ 上的对称函数，则k 是核函数当且仅当对于任意数据 $D=\{\boldsymbol{x_1},\boldsymbol{x_2},\ldots,\boldsymbol{x_m}\}$,核矩阵$K$ 总是半正定的:

$$
K = \begin{bmatrix}
k(\boldsymbol{x_1},\boldsymbol{x_1}) & \cdots & k(\boldsymbol{x_1},\boldsymbol{x_j}) & \cdots & k(\boldsymbol{x_1},\boldsymbol{x_m}) \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
k(\boldsymbol{x_i},\boldsymbol{x_1}) & \cdots & k(\boldsymbol{x_i},\boldsymbol{x_j}) & \cdots & k(\boldsymbol{x_i},\boldsymbol{x_m}) \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
k(\boldsymbol{x_m},\boldsymbol{x_1}) & \cdots & k(\boldsymbol{x_m},\boldsymbol{x_j}) & \cdots & k(\boldsymbol{x_m},\boldsymbol{x_m})
\end{bmatrix}
$$

通过上面的定理我们可以知道，只要一个对称函数所对应的核矩阵半正定，他就能作为核函数使用.事实上，对于一个半正定核矩阵，总能找到一个与之对应的映射ϕ.换言之，任何一个核函数都隐式的定义了一个称为"再生核希尔伯特空间"的特征空间。

由前面的公式我们可以知道特征空间的好坏对支持向量机的性能至关重要。需要注意的是，在不知道特征映射的形式时，我们并不知道什么样的核函数时合适的，而核函数也仅是隐式的定义了这个特征空间。于是，"核函数选择"成为支持向量机的最大变数。若核函数选择不合适，则意味着将样本映射到一个不合适的空间，很可能导致性能不佳。

常见的核函数有

线性核 $k(\boldsymbol{x_i},\boldsymbol{x_j}) = \boldsymbol{x_i^T x_j}$

多项式核 $k(\boldsymbol{x_i},\boldsymbol{x_j}) = (\boldsymbol{x_i^T x_j})^d$ $d \geq 1$为多项式的次数

高斯核 $k(\boldsymbol{x_i},\boldsymbol{x_j}) = \exp(-\frac{||\boldsymbol{x_i}-\boldsymbol{x_j}||^2}{2\sigma^2})$ $\sigma > 0$ 为高斯核的带宽(width)

拉普拉斯核 $k(\boldsymbol{x_i},\boldsymbol{x_j}) = \exp(-\frac{||\boldsymbol{x_i}-\boldsymbol{x_j}||}{\sigma})$ $\sigma > 0$

sigmod核 $k(\boldsymbol{x_i},\boldsymbol{x_j}) = \tanh(\beta \boldsymbol{x_i^T x_j} + \theta)$ $\tanh$ 为双曲正切函数，$\beta > 0, \theta < 0$

4.核方法

给定训练样本$\{(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),,\ldots,(\boldsymbol{x}_m,y_m)\}$,若不考虑偏移项$b$ 学得的模型总能表示成核函数的线性组合。有如下定理

定理2 （表示定理） 令$\mathbb{H}$ 为核函数k 对应的再生核希尔伯特空间 $\|h\|_{\mathbb{H}}$ 表示$\mathbb{H}$ 空间中关于h 的范数，对于任意单调递增函数$\Omega:[0,+\infty] \to \mathbb{R}$ 和任意非负损失函数$\ell:\mathbb{R}^m \to [0,+\infty]$ 优化问题:

$$\min_{h\in\mathbb{H}} F(h)=\Omega(\|h\|_{\mathbb{H}})+\ell(h(\boldsymbol{x}_1),h(\boldsymbol{x}_2),\ldots,h(\boldsymbol{x}_m)) \tag{20}$$

的解总写为：

$$h^*(\boldsymbol{x})=\sum_{i=1}^{m}\alpha_i k(\boldsymbol{x},\boldsymbol{x_i}) \tag{21}$$

表示定理对损失函数没有限制，对正则化项Ω 进要求单调递增，甚至不要求Ω 是凸函数。意味着对于一般的损失函数和正则化项，优化问题的最优解$h^*(\boldsymbol{x})$ 都可以表示为核函数的线性组合。

基于核函数的学习方法，统称为核方法。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。