小锅的机器学习笔记–支持向量机

数据:

$$
\begin{aligned}
&ll\\
i_t &= \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi})\\
f_t &= \sigma(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf})\\
g_t &= \tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg})\\
o_t &= \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho})\\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t\\
h_t &= o_t \odot \tanh(c_t)
\end{aligned}
$$

设样本为 (x_i, y_i)，$x_i \in R_p$，观察数据共有 n 个样本，分别为 $(x_1, y_i), \ldots, (x_n, y_n)$, 且

$$
y_i = \begin{cases} y_i = 1 & if \quad \text{第 } i \text{ 个样本为正类} \\ y_i = 0 & if \quad \text{第 } i \text{ 个样本为负类} \end{cases}
$$

SVM 的判别模型为:

$$
f(x_i) = sign(W^T x_i + b) \qquad sign(y) = \begin{cases} +1 & if \quad y \geq 0 \\ -1 & if \quad y < 0 \end{cases} \tag{1}
$$

式中 $W \in R_p$，$b \in R$ 为 SVM 模型要求解的参数。

硬间隔 SVM

硬间隔要求对每一个样本点分类正确, 那么有约束条件:

$$
s.t. \quad y_i \left(W^T x_i + b\right) > 0 \qquad i = 1, \ldots, n \tag{2}
$$

在高维空间中，设第 i 个样本点到超平面 $W^T x + b$ 的距离为:

$$
Dis_i = \frac{|W^T x_i + b|}{||W||_2} \tag{3}
$$

又因为约束条件 (2), 所以:

$$
Dis_i = \frac{y_i \left(W^T x_i + b\right)}{||W||_2} \tag{4}
$$

在机器学习中，追求的是泛化误差，而不是训练误差，要让泛化误差最大，也就是是要让离超平面最近的那个样本点离超平面的距离更最大化，也就是最大离超平面最近的那个点离超平面之间的函数几何间隔。

$$\begin{aligned} W,b &= \underset{W,b}{argmax} \underset{i=1,..,n}{min} Dis_i \\ &= \underset{W,b}{argmax} \frac{1}{||W||_2} \underset{i=1,..,n}{min} y_i \left(W^T x_i + b\right) \\ &= \underset{W,b}{argmax} \frac{1}{||W||_2} D \end{aligned}$$

其中 $D = \underset{i=1,..,n}{min} y_i \left(W^T x_i + b\right)$
综上，求解硬间隔的 SVM 就是求解一个优化问题，优化的约束与目标为:

$$\begin{cases} \underset{W,b}{max} \quad \frac{1}{||W||_2} D \\ s.t. \quad y_i \left(W^T x_i + b\right) \geq D \qquad i = 1, \ldots, n \end{cases} \tag{5}$$

$y_i \left(W^T x_i + b\right)$ 是点与函数之间的函数间隔，同时的让 W, b 增大 λ 倍变成 $\lambda W, \lambda b$，此时 $D, ||W||_2$ 也增加了 λ 倍，对我们的优化问题没有任何的影响，所以我们让 $D = 1$。最终优化问题可以写成:

$$\begin{cases} \underset{W,b}{max} \quad \frac{1}{||W||_2} \\ s.t. \quad 1 - y_i \left(W^T x_i + b\right) \leq 0 \qquad i = 1, \ldots, n \end{cases} \tag{6}$$

因为最大化 $\frac{1}{||W||_2}$ 和最小化 $\frac{W^T W}{2}$ 是等价的，所以可以继续写成

$$\begin{cases} \underset{W,b}{min} \quad \frac{W^T W}{2} \\ s.t. \quad 1 - y_i \left(W^T x_i + b\right) \leq 0 \qquad i = 1, \ldots, n \end{cases} \tag{7}$$

我们可以用拉格朗日乘子法改写上式成:

$$L = \frac{W^T W}{2} + \sum_{i=1}^{n} \lambda_i \left[1 - y_i \left(W^T x_i + b\right)\right] \quad \lambda_i \geq 0 \tag{8}$$

可以写成:

$$\begin{cases} \underset{W,b}{min} \underset{\lambda_i}{max} L \\ s.t. \quad \lambda_i \geq 0 \quad i = 1, \ldots n \end{cases} \tag{9}$$

证明不会，但是这样想，如果存在一个 x_i, y_i，使 $\left[1 - y_i \left(W^T x_i + b\right)\right] > 0$，那么在优化 $\underset{\lambda_i}{max} L$ 时就会让 $\lambda_i = +\infty$，此时 $\underset{W,b}{min} \underset{\lambda_i}{max} L = +\infty$，这样就没有任何意义了。
所以 $\left[1 - y_i \left(W^T x_i + b\right)\right] \leq 0 \quad i = 1, \ldots n$ 衡成立。

并且在优化 $\underset{\lambda_i}{max}\, L$ 时，会让 λ_i 全部为 0，那么我们最后就在求 $\underset{W,b}{min}\, \frac{W^TW}{2}$。
由于这是一个二次凸优化问题，存在强对偶关系，即:

$$\underset{W,b}{min}\, \underset{\lambda_i}{max}\, L = \underset{\lambda_i}{max}\, \underset{W,b}{min}\, L \tag{10}$$

首先我们求解 $\underset{W,b}{min}\, L$:

$$\begin{aligned} \frac{\partial L}{\partial W} &= W - \sum_{i=1}^{n} \lambda_i y_i x_i \\ \frac{\partial L}{\partial b} &= -\sum_{i=1}^{n} \lambda_i y_i \end{aligned}$$

令 $\frac{\partial L}{\partial W}, \frac{\partial L}{\partial b}$ 等于 0，有:。

$$\begin{cases} W = \sum_{i=1}^{n} \lambda_i y_i x_i \\ \sum_{i=1}^{n} \lambda_i y_i = 0 \end{cases} \tag{11}$$

将 (11) 带入 L 中可以得到:

$$\underset{W,b}{min}\, L = \frac{1}{2}(\sum_{i=1}^{n} \lambda_i y_i x_i)^T(\sum_{i=1}^{n} \lambda_i y_i x_i) + \sum_{i=1}^{n} \lambda_i \left[1 - y_i W^T x_i\right] \tag{12}$$

又因为:

$$\begin{aligned} &(\sum_{i=1}^{n} \lambda_i y_i x_i)^T(\sum_{i=1}^{n} \lambda_i y_i x_i) \\ &= \begin{bmatrix} \lambda_1 y_1, \lambda_2 y_2, \ldots, \lambda_n y_n \end{bmatrix} \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_n^T \end{bmatrix} \begin{bmatrix} x_1, x_2, \ldots, x_n \end{bmatrix} \begin{bmatrix} \lambda_1 y_1 \\ \lambda_2 y_2 \\ \vdots \\ \lambda_n y_n \end{bmatrix} \\ &= \begin{bmatrix} \lambda_1 y_1, \lambda_2 y_2, \ldots, \lambda_n y_n \end{bmatrix} \begin{bmatrix} x_1^T x_1, & x_1^T x_2, & \ldots, & x_1^T x_n \\ x_2^T x_1, & x_2^T x_2, & \ldots, & x_2^T x_n \\ \vdots & \vdots & \vdots & \vdots \\ x_n^T x_1, & x_n^T x_2, & \ldots, & x_n^T x_n \end{bmatrix} \begin{bmatrix} \lambda_1 y_1 \\ \lambda_2 y_2 \\ \vdots \\ \lambda_n y_n \end{bmatrix} \end{aligned}$$

这就是一个二次型，所以:

$$(\sum_{i=1}^{n} \lambda_i y_i x_i)^T(\sum_{i=1}^{n} \lambda_i y_i x_i) = \sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^T x_j \tag{13}$$

所以有:

$$\underset{W,b}{min}\, L = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^T x_j + \sum_{i=1}^{n} \lambda_i \left[1 - y_i W^T x_i\right] \tag{14}$$

最后我们再把 $W^T=\sum_{i=1}^n \lambda_i y_i x_i^T$ 带入:

$$\min_{W,b} L=\sum_{i=1}^{n}\lambda_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j y_i y_j x_i^T x_j \tag{15}$$

所以我们的最终需要优化的模型是:

$$\begin{cases}\max\limits_{\lambda_i} \quad \sum_{i=1}^n \lambda_i-\frac{1}{2}\sum_{i=1}^n\sum_{j=1}^n \lambda_i\lambda_j y_i y_j x_i^T x_j \\ \qquad s.t. \qquad \lambda_i\geq 0 \qquad i=1,\dots,n \\ \qquad\qquad\quad \sum_{i=1}^n \lambda_i y_i=0 \end{cases} \tag{16}$$

这个模型可以用 SMO 算法求解出 $(\lambda_1,\lambda_2,\dots,\lambda_n)$。剩下就是求解 W 和 b，因为 (9) 存在强对偶关系，所以必定满足 KKT 条件，即:

$$\begin{cases}\frac{\partial L}{\partial W}=0 \qquad \frac{\partial L}{\partial b}=0 \\ \lambda_i\left[1-y_i(W^T x_i+b)\right]=0 \qquad i=1,\dots,n \\ \lambda_i\geq 0 \qquad i=1,\dots,n \\ 1-y_i(W^T x_i+b)\leq 0 \end{cases} \tag{17}$$

当 $1-y_i(W^T x_i+b)<0$ 是，如果要使 $\lambda_i\left[1-y_i(W^T x_i+b)\right]=0$，那么此时一定有 $\lambda_i=0$。我们称满足 $1-y_i(W^T x_i+b)=0$ 的 (x_i,y_i) 为支持向量，对于支持向量，$\lambda_i\geq 0$。
支持向量一定存在，因为这个所谓的 1 是我们取的，本质上这个 1 的指的就是离分类超平面最近的那个向量与超平面的距离，所以我们至少存在一个支持向量 (离超平面最近的那个样本)。
设第 k 个样本为支持向量，$W=\sum_{i=1}^n \lambda_i y_i x_i$ 那么有:

$$y_k(W^T x_k+b)=1 \quad\Longleftrightarrow\quad b_k=y_k-(\sum_{i=1}^{n}\lambda_i y_i x_i^T)x_k$$

设支持向量的集合为 S，且对于非支持向量 x_f,y_f，它们对应的 λ_f 等于 0，那么:

$$\begin{cases} W \;=\; \sum_{x_k\in S}\lambda_k y_k x_k \\ b \;=\; \frac{1}{|S|}\sum_{x_k\in s}\left[y_k-(\sum_{i=1}^n \lambda_i y_i x_i^T)x_k\right] \end{cases} \tag{18}$$

我们可以发现无论是 W 还剩 b 都只与支持向量有关，这也是 SVM 被称为支持向量机的原因。

软间隔 SVM

通常，观测的数据存在噪声，也就是指有的正类样本被观察成了负类，或者原本数据就不是线性可分的，但是我就是要用线性分开，这种情况下硬间隔的 SVM 就不顶用

了，这个时候就需要软间隔的 SVM，软间隔的 SVM 和硬间隔的思想一样, 但是允许样本被分类错误，在 (7) 的基础上，记对样本 i 的分类错误损失为:

$$Loss_i = max\left\{0, 1 - y_i(W^T x_i + b)\right\} \tag{19}$$

如果 $1 - y_i(W^T x_i + b)$ 小于 0，那么这个时候样本 i 分类正确, 此时 $Loss_i$ 为 0，但是如果分类错误即 $1 - y_i(W^T x_i + b)$ 大于 0，此时 $Loss_i$ 就是损失。
对于第 i 个样本的约束条件改写为:

$$y_i\left(W^T x_i + b\right) \geq 1 - Loss_i \tag{20}$$

改成这样的原因是因为:

如果第 i 个样本分类真确，$Loss_i$ 为 0，此时约束条件不变。

如果第 i 个样本分类错误，$Loss_i = 1 - y_i(W^T x_i + b)$，此时约束不等式为衡等式，依旧满足。

所以 (7) 改写成:

$$\begin{cases} \underset{W,b}{min} \quad \dfrac{W^T W}{2} + C\sum_{i=1}^{n} Loss_i \\ s.t. \quad 1 - y_i\left(W^T x_i + b\right) - Loss_i \leq 0 \qquad i = 1, \ldots, n \end{cases} \tag{21}$$

C 为一个超参数，C 越大，软间隔 SVM 就越偏向把训练集中数据全部分类正确，C 越小，就偏向允许以训练集某些样本分类错误为代价去追去更大的间隔。

但是直接这样是没法搞的，我们引入松弛变量 θ_i，修改上式为:

$$\begin{cases} \underset{W,b,\theta_i}{min} \quad \dfrac{W^T W}{2} + C\sum_{i=1}^{n} \theta_i \\ s.t. \quad 1 - y_i\left(W^T x_i + b\right) - \theta_i \leq 0 \qquad i = 1, \ldots, n \\ \qquad \theta_i \geq 0 \end{cases} \tag{22}$$

当样本 i 分类正确时，有 $1 - y_i\left(W^T x_i + b\right) \leq 0$，我们追求目标最小化，此时会使 $\theta_i = 0$，此时无损失。

当样本 i 分类错误时，有 $1 - y_i\left(W^T x_i + b\right) > 0$，因为我们追求目标最小化，此时我们希望 θ_i 尽可能的小，但是需要满足 $1 - y_i\left(W^T x_i + b\right) - \theta_i \leq 0$，那么此时会使 $\theta_i = 1 - y_i\left(W^T x_i + b\right)$。

所以引入松弛变量 θ_i 后的优化目标 (22) 和原始 (21) 是一样的。
我们引入拉格朗日乘子，可以得到我们的优化目标函数为:

$$L = \frac{W^T W}{2} + C\sum_{i=1}^{n} \theta_i + \sum_{i=1}^{n} \lambda_i\left[1 - y_i\left(W^T x_i + b\right) - \theta_i\right] - \sum_{i=1}^{n} \mu_i \theta_i \tag{23}$$

其中 μ_i, θ_i 均大于等于 0。同硬间隔一样，软间隔可以写成:

$$\begin{cases} \underset{W,b,\theta_i}{min}\ \underset{\lambda_i}{max}\ L \\ s.t. \quad \lambda_i, \mu_i \geq 0 \quad i = 1, \dots n \end{cases} \tag{24}$$

由于是二次凸优化问题，所以存在强对偶关系, 上式可以写成:

$$\begin{cases} \underset{\lambda_i}{max}\ \underset{W,b,\theta_i}{min}\ L \\ s.t. \quad \lambda_i, \mu_i \geq 0 \quad i = 1, \dots n \end{cases} \tag{25}$$

对 L 分别求 W, b, θ_i 的偏导数并且令其均等于 0:

$$\begin{cases} \frac{\partial L}{\partial W} = W - \sum_{i=1}^{n} \lambda_i y_i x_i \\ \frac{\partial L}{\partial b} = -\sum_{i=1}^{n} \lambda_i y_i \\ \frac{\partial L}{\partial \theta_i} = C - \lambda_i - \mu_i \end{cases} \tag{26}$$

首先我们带入 $C = \lambda_i + \mu_i$ 进入 (23)，可以得到:

$$L = \frac{W^T W}{2} + \sum_{i=1}^{n} \lambda_i \left[1 - y_i \left(W^T x_i + b\right)\right]$$

带入 $W = \sum_{i=1}^{n} \lambda_i y_i x_i \quad \sum_{i=1}^{n} \lambda_i y_i = 0$，有:

$$L = \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^T x_j$$

因为 $\mu_i \geq 0, C = \lambda_i + \mu_i$，并且我们带入后的 L 没有 μ_i，所以我们相较于硬间隔的 SVM 多了一个约束条件:

$$0 \leq \lambda_i \leq C \qquad i = 1, \dots n \tag{27}$$

所以我们软间隔的优化函数可以写成:

$$\begin{cases} \underset{\lambda_i}{max} \quad \sum_{i=1}^{n} \lambda_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^T x_j \\ \qquad s.t. \qquad 0 \leq \lambda_i \leq C \qquad i = 1, \dots, n \\ \qquad\qquad \sum_{i=1}^{n} \lambda_i y_i = 0 \end{cases} \tag{28}$$

这个依旧可以利用 SMO 算法求解。

最后我们求 W, b：因为该问题存在强对偶关系，所以满足 KKT 条件:

$$\begin{cases} W = \sum_{i=1}^{n} \lambda_i y_i x_i \quad \sum_{i=1}^{n} \lambda_i y_i = 0 \quad C = \lambda_i + \mu_i \\ \mu_i \geq 0 \qquad \lambda_i \geq 0 \\ 1 - y_i(W^T x_i + b) - \theta_i \leq 0 \qquad \theta_i \geq 0 \\ \lambda_i \left[1 - y_i(W^T x_i + b) - \theta_i)\right] = 0 \qquad \mu_i \theta_i = 0 \end{cases} \tag{29}$$

显然有 $W=\sum_{i=1}^{n}\lambda_i y_i x_i$，其次若存在一个 $0<\lambda_k<C$，此时一定有 $u_k=C>0$，则 $\theta_k=0$，那么就是 $1-y_k(W^Tx_k+b)=0$，所以满足 $0<\lambda_k<C$ 的样本就是支持向量，所以:

$$b=y_k-W^Tx_i \tag{30}$$

核方法:

有些数据集就不是线性可分的，那么就算是使用了软间隔的 SVM 这个数据集也没法做，那么解决的方法是把低维度线性不可分的数据投影到高维度空间，那么就有可能可以分开。

设 R_1 是输入空间，R_2 是特征空间，如果存在一个从 R_1 到 R_2 的映射：

$$\phi(x):R_1\rightarrow R_2 \tag{31}$$

，使得对于所有的 $x,z\in R_1$。函数 $K(x,z)$ 都满足条件:

$$K(x,z)=\phi(x)\cdot\phi(z) \tag{32}$$

则称呼 $K(x,z)$ 为核函数。引入核函数后 SVM 的优化函数变成了：

$$\begin{cases} \underset{\lambda_i}{max} \quad \sum_{i=1}^{n}\lambda_i-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_i\lambda_j y_i y_j\phi(x_i^T)\phi(x_j) \\ \qquad s.t. \qquad 0\le\lambda_i\le C \qquad i=1,\dots,n \\ \qquad\qquad \sum_{i=1}^{n}\lambda_i y_i=0 \end{cases} \tag{33}$$

并且 W,b 变成了:

$$\begin{cases} W=\sum_{i=1}^{n}\lambda_i y_i\phi(x_i) \\ b=y_k-W^T\phi(x_i)=y_k-\sum_{j=1}^{n}\lambda_j y_j\phi(x_j^T)\phi(x_i)=y_k-\sum_{j=1}^{n}\lambda_j y_j K(x_i,x_j) \end{cases} \tag{34}$$

引入和函数后，W 就不能显式的求出来了，但是我们观察到，当我们要做预测 x_k 的时候需要计算 $W^T\phi(x_k)$，可以拆解为:

$$W^T\phi(x_k)=\sum_{i=1}^{n}\lambda_i y_i\phi(x_i)\phi(x_k)=\sum_{i=1}^{n}\lambda_i y_i K(x_i,x_k) \tag{35}$$