软间隔与正则化

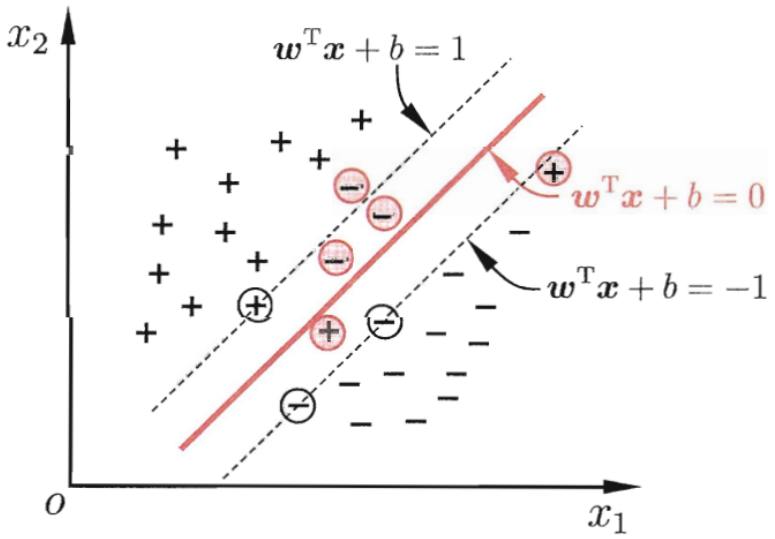
前言

我们已经处理了简单的对偶问题和非线性的处理问题，事实上我们就已经完成了 CS 229 所讲的内容了，但是为了深入，我们将讲解“软间隔与正则化”。

关于这部分内容仅参考了西瓜书，至于李航老师的《统计学习方法》（虽然是统计学习方法，但实际上就是个大大的机器学习），并没有提及.....所以很大程度上我们还是贴着教材走，当然也尽量多方求证。

其他的，就是关于“软间隔”的话题。我们在 SVM 开篇就吐槽：*“我们希望找到一个界限使得这个界限能够完完全全分开两侧的象棋，当然我们这里假设楚汉两方的棋是线性可分的（你看到了我标明了两处斜体红字。是的，我不可能一上来就谈软间隔和核函数下的 SVM，这既不合理、也不人性）。”* 这句话并不是白说的。我们要完完全全分开样本实际上有点“难为”模型了，毕竟真实世界里的样本数据可不一定完完全全就能分开。*那我们有没有办法整个一个宽松点的划分超平面，而不是模型真的“较真”过拟合造就的超平面呢？*

有的兄弟，有的。直接看图（来自西瓜书）：

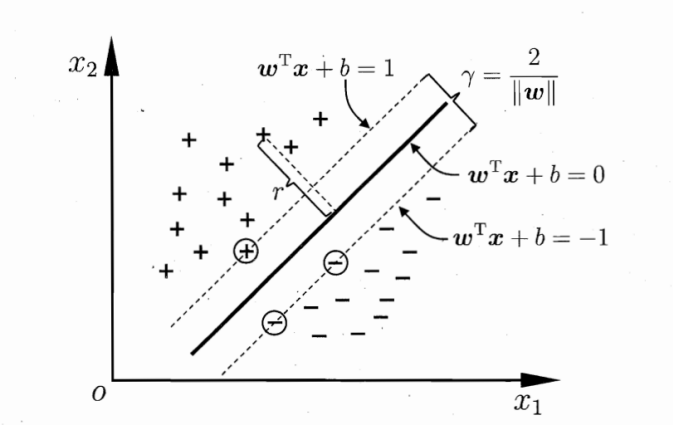


你看到了么？我们在原先的超平面周围一定距离下，整了个“隔离带”。这个“隔离带”代表了*即便有样本处于（错误分类到）这个区域也没关系*。这就是“软间隔(soft margin)”，而原来的那个叫做“硬间隔(hard margin)”。

软间隔下的 SVM

引出软间隔

当我们了解了软间隔，那我们可以对比一下原理的那张图：



我们只取了***在***$\mathbf{w}^{\top}\mathbf{x}+b=+1$和$\mathbf{w}^{\top}\mathbf{x}+b=-1$***上***的支持向量。而现在，只要向量***在上下边界内***，也就是满足$y_i(\mathbf{w}^{\top}x_i+b)\geq 1$即可！

注意，我说的是***在上下边界内！***

这是本质区别！

于是我们还是拿来这个公式：

$$y_i(\mathbf{w}^{\top}x_i+b)\geq 1,\quad (6.28)$$

还记得么？我们这个公式是由$\begin{cases}\mathbf{w}^{\top}x_i+b\geq+1,\ y_i=+1\\ \mathbf{w}^{\top}x_i+b\leq-1,\ y_i=-1\end{cases}$这个公式推导而来的。当然，在最大化间隔的同时，不满足约束条件的尽可能少，那么我们就可以把原来的优化问题：

$$\begin{aligned}&\min_{\mathbf{w},b}\quad \frac{1}{2}\|\mathbf{w}\|^2\\ &\text{s.t.}\quad y_i(\mathbf{w}^{\top}x_i+b)\geq 1,\quad i=1,2,...,m.\end{aligned},\quad (6.6)$$

改写成这个：

$$\begin{aligned}&\min_{\mathbf{w},b}\quad \frac{1}{2}\|\mathbf{w}\|^2+C\sum_{i=1}^{m}\ell_{0/1}\left(y_i(\mathbf{w}^{\top}x_i+b)-1\right)\\ &\text{s.t.}\quad y_i(\mathbf{w}^{\top}x_i+b)\geq 1,\quad i=1,2,...,m.\end{aligned},\quad (6.29)$$

你们肯定会疑惑，$C\sum_{i=1}^{m}\ell_{0/1}\left(y_i(\mathbf{w}^{\top}x_i+b)-1\right)$这一部分是干嘛的？解释一下，这部分项叫做“惩罚项”，也就是说模型错误越多，这惩罚项越大。我们的目标就是将这一项的值尽可能的减小。而其中的一些值：

- C是一个常数；
- $\ell_{0/1}(\cdot)$是“0/1损失函数”，形式如下

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

- $y_i(\mathbf{w}^{\top}x_i+b)-1$则是$y_i(\mathbf{w}^{\top}x_i+b)\geq 1,\quad (6.28)$移项得来

总体来看就是：当$1>y_i(\mathbf{w}^{\top}x_i+b)\geq 0$时，该向量在容忍区间内，属于“可容忍错误向量”，$y_i(\mathbf{w}^{\top}x_i+b)-1<0\Longrightarrow\ell_{0/1}\left(y_i(\mathbf{w}^{\top}x_i+b)-1\right)=1$，惩罚项增大1；当

$y_i(\mathbf{w}^\top x_i + b) \geq 1$ ***时，该向量要么不在区间内（大于 1 的时候），要么在区间边界属于支持向量（等于 1 的时候），此时*** $y_i(\mathbf{w}^\top x_i + b) - 1 \geq 0 \Longrightarrow \ell_{0/1}\big(y_i(\mathbf{w}^\top x_i + b) - 1\big) = 0$ ***，惩罚项不变。***

但是，由于 $\ell_{0/1}$ 是一个指示函数，非凸、非连续，数学性质不好，这就使得(6.29)不太好求。所以我们会想，能不能找个其他函数替代呢？有的有的，这类函数叫做“替代损失”(surrogate loss)。这类函数普遍具有很好的数学性质，比如凸、连续、可求导且是 $\ell_{0/1}$ 的上界。下面是一些替代损失函数（来自西瓜书）：

1. hinge损失(hinge loss)：$\ell_{hinge}(z) = \max(0, 1 - z);$ (6.31)
2. 指数损失(exponential loss)：$\ell_{exp}(z) = exp(-z);$ (6.32)
3. 对率损失(logistic loss)：$\ell_{log}(z) = log(1 + exp(-z));$ (6.33)

若采用 hinge 损失，则式(6.29)变成

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\max\big(0, 1 - y_i(\mathbf{w}^\top x_i + b)\big) \\ \text{s.t.} \quad & y_i(\mathbf{w}^\top x_i + b) \geq 1, \quad i = 1, 2, ..., m. \end{aligned}, \quad (6.34)$$

代换：引入“松弛变量”(slack variables) $\xi_i \geq 0$，可将(6.34)重写为

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i \\ \text{s.t.} \quad & y_i(\mathbf{w}^\top x_i + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \quad i = 1, 2, ..., m. \end{aligned}, \quad (6.35)$$

这就是软间隔下的支持向量机。但我们还需要做一些解释，其中关于 $y_i(\mathbf{w}^\top x_i + b) \geq 1 - \xi_i$ 这一项是我们需要注意的。***为什么要减去*** ξ_i ***？*** 其实就是为了处理那些在容忍区间内的向量。

当 $\xi_i = 0$ ***时，约束条件退化为*** $y_i(\mathbf{w}^\top x_i + b) \geq 1$ ***，意味着向量落在分类正确一侧，即样本点在支持向量机的分类间隔边界或者更外侧，是比较理想的分类情况，没有因为噪声等因素产生分类上的“松弛”。***

当 $\xi_i > 0$ ***时，约束条件为*** $y_i(\mathbf{w}^\top x_i + b) \geq 1 - \xi_i$ ***，可分为*** $1 > \xi_i > 0$ ***和*** $\xi_i \geq 1$ ***。***

- $1 > \xi_i > 0$ ***：样本点落在分类间隔内部，但仍在正确的类别一侧***
- $\xi_i \geq 1$ ***：样本点落在分类间隔之外，且被错误分类，因此*** $y_i(\mathbf{w}^\top x_i + b) < 1$

总的来说， ξ_i ***越大，样本偏离理想分类情况越远。***

软间隔 + 对偶问题 + 核函数（与 对偶问题、核函数 一致的推导思路）

我们得到了这个式子，依旧的可以看出来这是一个二次规划问题。于是，类似式(6.8)，通过拉格朗日乘子法直接得到式(6.35)的拉格朗日函数

$$
\begin{aligned}
L(\mathbf{w}, b, \alpha, \xi, \mu) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{m}\xi_i \\
+ \sum_{i=1}^{m}\alpha_i\big(1 - \xi_i - y_i(\mathbf{w}^\top x_i + b) - \sum_{i=1}^{m}\mu_i\xi_i\big)
\end{aligned}, \quad (6.36)
$$

其中 $\alpha_i \geq 0,\ \mu_i \geq 0$ 是拉格朗日乘子。

令 $L(\mathbf{w}, b, \alpha, \mu)$ 分别对 $\mathbf{w}, b, \xi_i$ 求偏导为 0 可得：

$$
\mathbf{w} = \sum_{i=1}^{m}\alpha_i y_i x_i, \quad (6.37)
$$

$$
0 = \sum_{i=1}^{m}\alpha_i y_i, \quad (6.38)
$$

$$
C = \alpha_i + \mu_i, \quad (6.3)
$$

将(6.37)-(6.39)代入式(6.39)即可得到式(6.35)的对偶问题

$$
\begin{aligned}
\max_{\alpha} \quad & \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j x_i^T x_j \\
\text{s.t.} \quad & \sum_{i=1}^{m}\alpha_i y_i = 0, \\
& 0 \leq \alpha_i \leq C, \quad i = 1, 2, \ldots, m.
\end{aligned}, \quad (6.40)
$$

将(6.40)与硬间隔下的对偶问题(6.11)对比可以看出，两者唯一的差别就在于对偶变量的约束不同：前者是 $C \geq \alpha_i \geq 0$，后者是 $\alpha_i \geq 0$。于是，采用 对偶问题 当中的 SMO 算法同样可求解式(6.40)；引入核函数后得到

$$
\begin{aligned}
\max_{\alpha} \quad & \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \kappa(x_i, x_j) \\
\text{s.t.} \quad & \sum_{i=1}^{m}\alpha_i y_i = 0, \\
& 0 \leq \alpha_i \leq C, \quad i = 1, 2, \ldots, m.
\end{aligned}
$$

而对于 KKT，我们类似(6.13)，对软间隔支持向量机，KKT 条件有

$$
\begin{cases}
\alpha_i \geq 0, \quad \mu_i \geq 0, ① \\
y_i f(x_i) - 1 + \xi_i \geq 0, ② \\
\alpha_i(y_i f(x_i) - 1 + \xi_i) = 0, ③ \\
\xi_i \geq 0, \quad \mu_i \xi_i = 0, ④
\end{cases}, \quad (6.41)
$$

对任意训练样本(x_i, y_i)，总有$\alpha_i = 0$或$y_i f(x_i) = 1 - \xi_i$。若$\alpha_i = 0$，则该样本不会对$f(\mathbf{x})$有任何影响；若$\alpha_i > 0$，则必有$y_i f(x_i) = 1 - \xi_i$，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$\mu_i = 0$，则有$\mu_i = 0$，此时若$\xi_i \leq 1$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。由此可以看出，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（某个数据集或矩阵中，大部分元素都是零或接近于零的情况）。

其他（来自西瓜书）

如果使用其他替代损失函数呢？如果使用对率损失函数来替代(6.29)中的 0/1 损失函数，则几乎得到了几率回归模型。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，从上图可看出，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*。

正则化

这一路走来的推导，我们无非就是在$\frac{1}{2}\|\mathbf{w}\|^2$周围做做手脚，甚至是上边所说的“替换其他损失函数变换得到其他学习模型”...那么有没有一种统一的表述方式呢？有的兄弟，有的。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$\sum_{i=1}^{m} \ell(f(x_i), y_i)$来表示训练集上的误差（损失期望）*，所以直接优化成更一般的形式，有：

$$\min_f \Omega(f) + C \sum_{i=1}^{m} \ell(f(x_i), y_i), \quad (6.42)$$

其中$\Omega(f)$被称为“结构风险”(structural risk)，描述一个模型f的某些性质（比如 SVM 的间隔大小）；第二项$\sum_{i=1}^{m} \ell(f(x_i), y_i)$被称为“经验风险”(empirical risk)，用于描述模型与训练数据的契合程度（其实这一项也可以叫平均损失，是损失函数在整个训练集上的期望）；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，$\Omega(f)$被称为正则化项，C则是称为正则化常数。L_p 范数(norm)是常用的正则化项，其中L_2范数$\|\mathbf{w}\|_2$倾向于$\mathbf{w}$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$\|w\|_0$和L_1范数$\|w\|_1$则倾向于$\mathbf{w}$的分量尽量稀疏，即非零分量个数尽量少。

注：正则化可理解为一种“罚函数法”，即对不希望得到的结果施以惩罚，从而使得优化过程趋向于希望目标，从贝叶斯估计的角度来看，正则化项可认为是提供了模型的先验概率。