从零构建支持向量机 (SVM)

张皓

南京大学软件新技术国家重点实验室

zhangh0214@gmail.com

摘要

支持向量机 (SVM) 是一个非常经典且高效的分类模型. 但是, 支持向量机中涉及许多复杂的数学推导, 并需要比较强的凸优化基础, 使得有些初学者虽下大量时间和精力研读, 但仍一头雾水, 最终对其望而却步. 本文旨在从零构建支持向量机, 涵盖从思想到形式化, 再简化, 最后实现的完整过程, 并展现其完整思想脉络和所有公式推导细节. 本文力图做到逻辑清晰而删繁就简, 避免引入不必要的概念, 记号等. 此外, 本文并不需要读者有凸优化的基础, 以减轻读者的负担. 对于用到的优化技术, 在文中均有其介绍.

尽管现在深度学习十分流行, 了解支持向量机的原理, 对想法的形式化, 简化, 及一步步使模型更一般化的过程, 及其具体实现仍然有其研究价值. 另一方面, 支持向量机仍有其一席之地. 相比深度神经网络, 支持向量机特别擅长于特征维数多于样本数的情况, 而小样本学习至今仍是深度学习的一大难题.

1 线性二分类模型

给定一组数据 $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$, 其中 $\boldsymbol{x}_i \in \mathbb{R}^d, y \in \{-1, 1\}$, 二分类任务的目标是希望从数据中学得一个假设函数 $h\colon \mathbb{R} \to \{-1, 1\}$, 使得 $h(\boldsymbol{x}_i) = y_i$, 即

$$h(\boldsymbol{x}_i) = \begin{cases} 1 & 若\ y_i = 1\,; \\ -1 & 若\ y_i = -1\,. \end{cases} \tag{1}$$

用一个更简洁的形式表示是

$$\forall i.\ y_i h(\boldsymbol{x}_i) = 1\,. \tag{2}$$

更进一步, 线性二分类模型认为假设函数的形式是基于对特征 $\boldsymbol{x}_i$ 的线性组合, 即

$$h(\boldsymbol{x}_i) := \operatorname{sign}(\boldsymbol{w}^\top \boldsymbol{x}_i + b)\,, \tag{3}$$

其中 $\boldsymbol{w}_i \in \mathbb{R}^d, b \in \mathbb{R}$.

定理 1. *线性二分类模型的目标是找到一组合适的参数 $(\boldsymbol{w}, b)$, 使得*

$$\forall i.\ y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) > 0\,. \tag{4}$$

即, 线性二分类模型希望在特征空间找到一个划分超平面, 将属于不同标记的样本分开.

证明.

$$y_i h(\boldsymbol{x}_i) = 1 \Leftrightarrow y_i \operatorname{sign}(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = 1 \Leftrightarrow y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) > 0\,. \tag{5}$$

2 线性支持向量机

线性支持向量机 (SVM) [4]也是一种线性二分类模型, 也需要找到满足定理 1约束的划分超平面, 即 $(\boldsymbol{w}, b)$. 由于能将样本分开的超平面可能有很多, SVM 进一步希望找到离各样本都比较远的划分超平面.

当面对对样本的随机扰动时, 离各样本都比较远的划分超平面对扰动的容忍能力比较强, 即不容易因为样本的随机扰动使样本穿越到划分超平面的另外一侧而产生分类错误. 因此, 这样的划分超平面对样本比较稳健, 不容易过拟合. 另一方面, 离各样本都比较远的划分

超平面不仅可以把正负样本分开, 还可以以比较大的确信度将所有样本分开, 包括难分的样本, 即离划分超平面近的样本.

2.1 间隔

在支持向量机中, 我们用间隔 (margin) 刻画划分超平面与样本之间的距离. 在引入间隔之前, 我们需要先知道如何计算空间中点到平面的距离.

引理 2. $\mathbb{R}^d$ 空间中某点 $\boldsymbol{p} \in \mathbb{R}^d$ 到超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ 的距离为

$$\frac{1}{\|\boldsymbol{w}\|} |\boldsymbol{w}^\top \boldsymbol{p} + b| \,. \tag{6}$$

证明. 假设 $\boldsymbol{x}_1, \boldsymbol{x}_2$ 是该超平面上两点, 则

$$\boldsymbol{w}^\top (\boldsymbol{x}_1 - \boldsymbol{x}_2) = \boldsymbol{w}^\top \boldsymbol{x}_1 - \boldsymbol{w}^\top \boldsymbol{x}_2 = (-b) - (-b) = 0 \,, \tag{7}$$

即 $\boldsymbol{w} \perp (\boldsymbol{x}_1 - \boldsymbol{x}_2)$. 又因为 $\boldsymbol{x}_1 - \boldsymbol{x}_2$ 与该超平面平行, 则 $\boldsymbol{w}$ 与该超平面垂直. 点 $\boldsymbol{p}$ 到该超平面的距离等于 $\boldsymbol{p}$ 与超平面上某点 $\boldsymbol{x}$ 连线向超平面法向量 (即, $\boldsymbol{w}$) 的投影:

$$\begin{aligned}
\operatorname{proj}_{\boldsymbol{w}}(\boldsymbol{p} - \boldsymbol{x}) &= \|\boldsymbol{p} - \boldsymbol{x}\| \cdot |\cos\langle \boldsymbol{w}, \boldsymbol{p} - \boldsymbol{x} \rangle| \\
&= \|\boldsymbol{p} - \boldsymbol{x}\| \cdot \frac{|\boldsymbol{w}^\top (\boldsymbol{p} - \boldsymbol{x})|}{\|\boldsymbol{w}\| \|\boldsymbol{p} - \boldsymbol{x}\|} \\
&= \frac{1}{\|\boldsymbol{w}\|} |\boldsymbol{w}^\top \boldsymbol{p} - \boldsymbol{w}^\top \boldsymbol{x}| \\
&= \frac{1}{\|\boldsymbol{w}\|} |\boldsymbol{w}^\top \boldsymbol{p} + b| \,.
\end{aligned} \tag{8}$$

定义 1 (间隔 γ)**.** 间隔表示距离划分超平面最近的样本到划分超平面距离的两倍, 即

$$\gamma := 2 \min_i \frac{1}{\|\boldsymbol{w}\|} |\boldsymbol{w}^\top \boldsymbol{x}_i + b| \,.$$

也就是说, 间隔表示划分超平面到属于不同标记的最近样本的距离之和.

定理 3. 线性支持向量机的目标是找到一组合适的参数 $(\boldsymbol{w}, b)$, 使得

$$\begin{aligned}
\max_{\boldsymbol{w}, b} \min_i \quad & \frac{2}{\|\boldsymbol{w}\|} |\boldsymbol{w}^\top \boldsymbol{x}_i + b| \\
\text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) > 0, \quad i = 1, 2, \ldots, m \,.
\end{aligned} \tag{9}$$

即, 线性支持向量机希望在特征空间找到一个划分超平面, 将属于不同标记的样本分开, 并且该划分超平面距离各样本最远.

证明. 带入间隔定义即得.

2.2 线性支持向量机基本型

定理 3描述的优化问题十分复杂, 难以处理. 为了能在现实中应用, 我们希望能对其做一些简化, 使其变为可以求解的, 经典的凸二次规划 (QP) 问题.

定义 2 (凸二次规划)**.** 凸二次规划的优化问题是指目标函数是凸二次函数, 约束是线性约束的一类优化问题.

$$\begin{aligned}
\min_{\boldsymbol{u}} \quad & \frac{1}{2} \boldsymbol{u}^\top \boldsymbol{Q} \boldsymbol{u} + \boldsymbol{t}^\top \boldsymbol{u} \\
\text{s.t.} \quad & \boldsymbol{c}_i^\top \boldsymbol{u} \geq d_i, \quad i = 1, 2, \ldots, m \,.
\end{aligned} \tag{10}$$

引理 4. 若 $(\boldsymbol{w}^\star, b^\star)$ 是定理 3优化问题的解, 那么对任意 $r > 0$, $(r\boldsymbol{w}^\star, rb^\star)$ 仍是该优化问题的解.

证明.

$$\frac{2}{\|r\boldsymbol{w}^\star\|} |(r\boldsymbol{w}^\star)^\top \boldsymbol{x}_i + rb^\star| = \frac{2}{\|\boldsymbol{w}^\star\|} |\boldsymbol{w}^{\star\top} \boldsymbol{x}_i + b^\star| \,, \tag{11}$$

$$y_i((r\boldsymbol{w}^\star)^\top \boldsymbol{x}_i + rb^\star) > 0 \Leftrightarrow y_i(\boldsymbol{w}^{\star\top} \boldsymbol{x}_i + b^\star) > 0 \,. \tag{12}$$

由于对 $(\boldsymbol{w}, b)$ 的放缩不影响解, 为了简化优化问题, 我们约束 $(\boldsymbol{w}, b)$ 使得

$$\min_i |\boldsymbol{w}^\top \boldsymbol{x}_i + b| = 1 \,. \tag{13}$$

定理 5 (线性支持向量机基本型)**.** 定理 3描述的线性支持向量机的优化问题等价于找到一组合适的参数 $(\boldsymbol{w}, b)$, 使得

$$\begin{aligned}
\min_{\boldsymbol{w}, b} \quad & \frac{1}{2} \boldsymbol{w}^\top \boldsymbol{w} \\
\text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1, \quad i = 1, 2, \ldots, m \,.
\end{aligned} \tag{14}$$

证明. 对约束项, 我们采用反证法. 假设最优值 $(\boldsymbol{w}^\star, b^\star)$ 处等号不成立, 即 $\min_i y_i(\boldsymbol{w}^{\star\top} \boldsymbol{x}_i + b^\star) > 1$. 此时存在 $(r\boldsymbol{w}, rb)$, 其中 $0 < r < 1$, 使得 $\min_i y_i((r\boldsymbol{w})^\top \boldsymbol{x}_i + rb) = 1$, 且 $\frac{1}{2}\|r\boldsymbol{w}\|^2 < \frac{1}{2}\|\boldsymbol{w}\|^2$. 说明 $(\boldsymbol{w}^\star, r^\star)$ 不是最优值, 与假设矛盾. 因此, 公式 14等价于

$$\begin{aligned}
\min_{\boldsymbol{w}, b} \quad & \frac{1}{2} \boldsymbol{w}^\top \boldsymbol{w} \\
\text{s.t.} \quad & \min_i y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = 1 \,.
\end{aligned} \tag{15}$$

优化目标等价于

$$\begin{aligned}
\operatorname*{arg\,min}_{\boldsymbol{w}, b} \frac{1}{2} \boldsymbol{w}^\top \boldsymbol{w} &= \operatorname*{arg\,min}_{\boldsymbol{w}, b} \frac{1}{2} \|\boldsymbol{w}\| \\
&= \operatorname*{arg\,max}_{\boldsymbol{w}, b} \frac{2}{\|\boldsymbol{w}\|} \cdot 1
\end{aligned}$$

$$= \arg\max_{\boldsymbol{w},b} \left(\min_i \frac{2}{\|\boldsymbol{w}\|} y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \right)$$

$$= \arg\max_{\boldsymbol{w},b} \left(\min_i \frac{2}{\|\boldsymbol{w}\|} |\boldsymbol{w}^\top \boldsymbol{x}_i + b| \right) \tag{16}$$

推论 6. *线性支持向量机基本型中描述的优化问题属于二次规划问题，包括 $d+1$ 个优化变量，m 项约束.*

证明. 令

$$\boldsymbol{u} := \begin{bmatrix} \boldsymbol{w} \\ b \end{bmatrix}, \ \boldsymbol{Q} := \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & 0 \end{bmatrix}, \ \boldsymbol{t} := \boldsymbol{0}\,, \tag{17}$$

$$\boldsymbol{c}_i := y_i \begin{bmatrix} \boldsymbol{x}_i \\ 1 \end{bmatrix}, \ d_i := 1\,, \tag{18}$$

代入公式 10即得.

3 对偶问题

现在，我们可以通过调用现成的凸二次规划软件包来求解定理 5 描述的优化问题. 不过，通过借助拉格朗日 (Lagrange) 函数和对偶 (dual) 问题，我们可以将问题更加简化.

3.1 拉格朗日函数与对偶形式

构造拉格朗日函数是求解带约束优化问题的重要方法.

定义 3 (拉格朗日函数)**.** 对于优化问题

$$\begin{aligned} \min_{\boldsymbol{u}} \quad & f(\boldsymbol{u}) \\ \text{s.t.} \quad & g_i(\boldsymbol{u}) \le 0, \quad i = 1, 2, \ldots, m\,, \\ & h_j(\boldsymbol{u}) = 0, \quad j = 1, 2, \ldots, n\,, \end{aligned} \tag{19}$$

定义其拉格朗日函数为

$$\mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\beta}) := f(\boldsymbol{u}) + \sum_{i=1}^m \alpha_i g_i(\boldsymbol{u}) + \sum_{j=1}^n \beta_j h_j(\boldsymbol{u})\,, \tag{20}$$

其中 $\alpha_i \ge 0$.

引理 7. *公式 19描述的优化问题等价于*

$$\begin{aligned} \min_{\boldsymbol{u}} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \quad & \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \\ \text{s.t.} \quad & \alpha_i \ge 0, \quad i = 1, 2, \ldots, m\,. \end{aligned} \tag{21}$$

证明.

$$\begin{aligned} & \min_{\boldsymbol{u}} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \\ & = \min_{\boldsymbol{u}} \left(f(\boldsymbol{u}) + \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \left(\sum_{i=1}^m \alpha_i g_i(\boldsymbol{u}) + \sum_{j=1}^n \beta_j h_j(\boldsymbol{u}) \right) \right) \\ & = \min_{\boldsymbol{u}} \left(f(\boldsymbol{u}) + \begin{cases} 0 & \text{若 } \boldsymbol{u} \text{ 满足约束}; \\ \infty & \text{否则} \end{cases} \right) \\ & = \min_{\boldsymbol{u}} f(\boldsymbol{u}), \text{ 且 } \boldsymbol{u} \text{ 满足约束}, \end{aligned} \tag{22}$$

其中，当 g_i 不满足约束时，即 $g_i(\boldsymbol{u}) > 0$，我们可以取 $\alpha_i = \infty$，使得 $\alpha_i g_i(\boldsymbol{u}) = \infty$；当 h_j 不满足约束时，即 $h_j(\boldsymbol{u}) \neq 0$，我们可以取 $\beta_j = \text{sign}(h_j(\boldsymbol{u}))\infty$，使得 $\beta_j h_j(\boldsymbol{u}) = \infty$. 当 $\boldsymbol{u}$ 满足约束时，由于 $\alpha_i \ge 0$，$g_i(\boldsymbol{u}) \le 0$，则 $\alpha_i g_i(\boldsymbol{u}) \le 0$. 因此 $\alpha_i g_i(\boldsymbol{u})$ 最大值为 0.

推论 8 (KKT 条件)**.** *公式 21描述的优化问题在最优值处必须满足如下条件.*

- *主问题可行: $g_i(\boldsymbol{u}) \le 0, h_i(\boldsymbol{u}) = 0$;*
- *对偶问题可行: $\alpha_i \ge 0$;*
- *互补松弛 (complementary slackness): $\alpha_i g_i(\boldsymbol{u}) = 0$.*

证明. 由引理 7可知，$\boldsymbol{u}$ 必须满足约束，即主问题可行. 对偶问题可行是公式 21描述的优化问题的约束项. $\alpha_i g_i(\boldsymbol{u}) = 0$ 是在主问题和对偶问题都可行的条件下的最大值.

定义 4 (对偶问题)**.** 定义公式 19描述的优化问题的对偶问题为

$$\begin{aligned} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \min_{\boldsymbol{u}} \quad & \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \\ \text{s.t.} \quad & \alpha_i \ge 0, \quad i = 1, 2, \ldots, m\,. \end{aligned} \tag{23}$$

引理 9. *对偶问题是主 (primal) 问题的下界，即*

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \min_{\boldsymbol{u}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \le \min_{\boldsymbol{u}} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\beta})\,. \tag{24}$$

证明. 对任意 $(\boldsymbol{\alpha}', \boldsymbol{\beta}')$，$\min_{\boldsymbol{u}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}', \boldsymbol{\beta}') \le \min_{\boldsymbol{u}} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\beta})$. 当 $(\boldsymbol{\alpha}', \boldsymbol{\beta}') = \max_{\boldsymbol{\alpha}',\boldsymbol{\beta}'} \min_{\boldsymbol{u}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}', \boldsymbol{\beta}')$ 时，该式仍然成立，即 $\max_{\boldsymbol{\alpha}',\boldsymbol{\beta}'} \min_{\boldsymbol{u}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}', \boldsymbol{\beta}') \le \min_{\boldsymbol{u}} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\beta})$.

引理 10 (Slater 条件)**.** *当主问题为凸优化问题，即 f 和 g_i 为凸函数，h_j 为仿射函数，且可行域中至少有一点使不等式约束严格成立时，对偶问题等价于原问题.*

证明. 此证明已超出本文范围, 感兴趣的读者可参考 [2].

推论 11. *线性支持向量机满足 Slater 条件.*

证明. $\frac{1}{2}\boldsymbol{w}^\top\boldsymbol{w}$ 和 $1-y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b)$ 均为凸函数.

3.2 线性支持向量机对偶型

线性支持向量机的拉格朗日函数为

$$\mathcal{L}(\boldsymbol{w},b,\boldsymbol{\alpha}) := \frac{1}{2}\boldsymbol{w}^\top\boldsymbol{w}+\sum_{i=1}^{m}\alpha_i(1-y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b)) \,. \quad (25)$$

其对偶问题为

$$\begin{aligned} \max_{\boldsymbol{\alpha}}\min_{\boldsymbol{w},b} \quad & \frac{1}{2}\boldsymbol{w}^\top\boldsymbol{w}+\sum_{i=1}^{m}\alpha_i(1-y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b)) \quad (26)\\ \text{s.t.} \quad & \alpha_i \geq 0, \quad i=1,2,\ldots,m \,. \end{aligned}$$

定理 12 (线性支持向量机对偶型)**.** *线性支持向量机的对偶问题等价于找到一组合适的参数 $\boldsymbol{\alpha}$, 使得*

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^\top\boldsymbol{x}_j-\sum_{i=1}^{m}\alpha_i \quad (27)\\ \text{s.t.} \quad & \sum_{i=1}^{m}\alpha_i y_i=0,\\ & \alpha_i \geq 0, \quad i=1,2,\ldots,m \,. \end{aligned}$$

证明. 因为公式 26内层对 $(\boldsymbol{w},b)$ 的优化属于无约束优化问题, 我们可以通过令偏导等于零的方法得到 $(\boldsymbol{w},b)$ 的最优值.

$$\frac{\partial\mathcal{L}}{\partial\boldsymbol{w}}=\mathbf{0}\Rightarrow\boldsymbol{w}=\sum_{i=1}^{m}\alpha_i y_i\boldsymbol{x}_i \,, \quad (28)$$

$$\frac{\partial\mathcal{L}}{\partial b}=0\Rightarrow\sum_{i=1}^{m}\alpha_i y_i=0 \,. \quad (29)$$

将其代入公式 26, 消去 $(\boldsymbol{w},b)$, 即得.

推论 13. *线性支持向量机对偶型中描述的优化问题属于二次规划问题, 包括 m 个优化变量, $m+2$ 项约束.*

证明. 令

$$\boldsymbol{u}:=\boldsymbol{\alpha},\ \boldsymbol{Q}:=[y_i y_j\boldsymbol{x}_i^\top\boldsymbol{x}_j]_{m\times m},\ \boldsymbol{t}:=-\mathbf{1} \,, \quad (30)$$

$$\boldsymbol{c}_i:=\mathbf{e}_i,\ d_i:=0, \quad i=1,2,\ldots,m \,, \quad (31)$$

$$\boldsymbol{c}_{m+1}:=[y_1\ y_2\ \cdots\ y_m]^\top,\ d_{m+1}:=0 \,, \quad (32)$$

$$\boldsymbol{c}_{m+2}:=-[y_1\ y_2\ \cdots\ y_m]^\top,\ d_{m+2}:=0 \,, \quad (33)$$

代入公式 10即得. 其中, $\mathbf{e}_i$ 是第 i 位置元素为 1, 其余位置元素为 0 的单位向量. 我们需要通过两个不等式约束 $\boldsymbol{c}_{m+1}^\top\boldsymbol{u}\leq d_{m+1}$ 和 $\boldsymbol{c}_{m+2}^\top\boldsymbol{u}\leq d_{m+2}$ 来得到一个等式约束.

3.3 支持向量

定理 14 (线性支持向量机的 KKT 条件)**.** *线性支持向量机的 KKT 条件如下.*

- *主问题可行: $1-y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b)\leq 0$;*
- *对偶问题可行: $\alpha_i\geq 0$;*
- *互补松弛: $\alpha_i(1-y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b))=0$.*

证明. 令

$$\boldsymbol{u}:=\begin{bmatrix}\boldsymbol{w}\\ b\end{bmatrix},\ g_i(\boldsymbol{u}):=1-y_i\begin{bmatrix}\boldsymbol{x}_i\\ 1\end{bmatrix}^\top\boldsymbol{u} \,, \quad (34)$$

代入引理 8即得.

定义 5 (支持向量)**.** 对偶变量 $\alpha_i>0$ 对应的样本.

引理 15. *线性支持向量机中, 支持向量是距离划分超平面最近的样本, 落在最大间隔边界上.*

证明. 由线性支持向量机的 KKT 条件可知, $\alpha_i(1-y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b))=0$. 当 $\alpha_i>0$ 时, $1-y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b)=0$. 即 $y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b)=1$.

定理 16. *支持向量机的参数 $(\boldsymbol{w},b)$ 仅由支持向量决定, 与其他样本无关.*

证明. 由于对偶变量 $\alpha_i>0$ 对应的样本是支持向量,

$$\begin{aligned} \boldsymbol{w} &= \sum_{i=1}^{m}\alpha_i y_i\boldsymbol{x}_i\\ &= \sum_{i:\ \alpha_i=0}^{m} 0\cdot y_i\boldsymbol{x}_i+\sum_{i:\ \alpha_i>0}^{m}\alpha_i y_i\boldsymbol{x}_i\\ &= \sum_{i\in SV}\alpha_i y_i\boldsymbol{x}_i \,, \end{aligned} \quad (35)$$

其中 SV 代表所有支持向量的集合. b 可以由互补松弛松弛算出. 对于某一支持向量 $\boldsymbol{x}_s$ 及其标记 y_s, 由于 $y_s(\boldsymbol{w}^\top\boldsymbol{x}_s+b)=1$, 则

$$b=y_s-\boldsymbol{w}^\top\boldsymbol{x}_s=y_s-\sum_{i\in SV}\alpha_i y_i\boldsymbol{x}_i^\top\boldsymbol{x}_s \,. \quad (36)$$

实践中, 为了得到对 b 更稳健的估计, 通常使用对所有支持向量求解得到 b 的平均值.

推论 17. 线性支持向量机的假设函数可表示为

$$h(\boldsymbol{x}) = \operatorname{sign}\left(\sum_{i \in SV} \alpha_i y_i \boldsymbol{x}_i^\top \boldsymbol{x} + b\right). \tag{37}$$

证明. 代入公式 35即得.

4 核函数

至此, 我们都是假设训练样本是线性可分的. 即, 存在一个划分超平面能将属于不同标记的训练样本分开. 但在很多任务中, 这样的划分超平面是不存在的. 支持向量机通过核技巧 (kernel trick) 来解决样本不是线性可分的情况 [1].

4.1 非线性可分问题

既然在原始的特征空间 $\mathbb{R}^d$ 不是线性可分的, 支持向量机希望通过一个映射 $\phi\colon \mathbb{R}^d \to \mathbb{R}^{\tilde{d}}$, 使得数据在新的空间 $\mathbb{R}^{\tilde{d}}$ 是线性可分的.

引理 18. 当 d 有限时, 一定存在 $\tilde{d}$, 使得样本在空间 $\mathbb{R}^{\tilde{d}}$ 中线性可分.

证明. 此证明已超出本文范围, 感兴趣的读者可参考计算学习理论中打散 (shatter) 的相应部分 [16].

令 $\boldsymbol{\phi}(\boldsymbol{x})$ 代表将样本 $\boldsymbol{x}$ 映射到 $\mathbb{R}^{\tilde{d}}$ 中的特征向量, 参数 $\boldsymbol{w}$ 的维数也要相应变为 $\tilde{d}$ 维. 则支持向量机的基本型和对偶型相应变为:

$$\begin{aligned} \min_{\boldsymbol{w}, b} \quad & \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{w} \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b) \geq 1, \quad i = 1, 2, \ldots, m; \end{aligned} \tag{38}$$

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j) - \sum_{i=1}^m \alpha_i \\ \text{s.t.} \quad & \sum_{i=1}^m \alpha_i y_i = 0, \\ & \alpha_i \geq 0, \quad i = 1, 2, \ldots, m. \end{aligned} \tag{39}$$

其中, 基本型对应于 $\tilde{d} + 1$ 个优化变量, m 项约束的二次规划问题; 对偶型对应于 m 个优化变量, $m + 2$ 项约束的二次规划问题.

4.2 核技巧

注意到, 在支持向量机的对偶型中, 被映射到高维的特征向量总是以成对内积的形式存在, 即 $\boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j)$. 如果先计算特征在 $\mathbb{R}^{\tilde{d}}$ 空间的映射, 再计算内积, 复杂度是 $\mathcal{O}(\tilde{d})$. 当特征被映射到非常高维的空间, 甚至是无穷维空间时, 这将会是沉重的存储和计算负担.

核技巧旨在将特征映射和内积这两步运算压缩为一步, 并且使复杂度由 $\mathcal{O}(\tilde{d})$ 降为 $\mathcal{O}(d)$. 即, 核技巧希望构造一个核函数 $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$, 使得

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j), \tag{40}$$

并且 $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ 的计算复杂度是 $\mathcal{O}(d)$.

引理 19. 映射

$$\phi\colon x \mapsto \exp(-x^2)\begin{bmatrix} 1 \\ \sqrt{\frac{2}{1}}x \\ \sqrt{\frac{2^2}{2!}}x^2 \\ \vdots \end{bmatrix} \tag{41}$$

对应于核函数

$$\kappa(x_i, x_j) := \exp(-(x_i - x_j)^2). \tag{42}$$

证明.

$$\begin{aligned} \kappa(x_i, x_j) &= \exp(-(x_i - x_j)^2) \\ &= \exp(-x_i^2)\exp(-x_j^2)\exp(2x_i x_j) \\ &= \exp(-x_i^2)\exp(-x_j^2)\sum_{k=0}^\infty \frac{(2x_i x_j)^k}{k} \\ &= \sum_{k=0}^\infty \left(\exp(-x_i^2)\sqrt{\frac{2^k}{k!}}x_i^k\right)\left(\exp(-x_j^2)\sqrt{\frac{2^k}{k!}}x_j^k\right) \\ &= \boldsymbol{\phi}(x_i)^\top \boldsymbol{\phi}(x_j). \end{aligned} \tag{43}$$

4.3 核函数选择

通过向高维空间映射及核技巧, 我们可以高效地解决样本非线性可分问题. 但面对一个现实任务, 我们很难知道应该具体向什么样的高维空间映射, 即应该选什么样的核函数, 而核函数选择的适合与否直接决定整体的性能.

表 1列出了几种常用的核函数. 通常, 当特征维数 d 超过样本数 m 时 (文本分类问题通常是这种情况), 使

用线性核; 当特征维数 d 比较小. 样本数 m 中等时, 使用 RBF 核; 当特征维数 d 比较小. 样本数 m 特别大时, 支持向量机性能通常不如深度神经网络.

除此之外, 用户还可以根据需要自定义核函数, 但需要满足 Mercer 条件 [5].

定理 20 (Mercer 条件)**.** 核函数 $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ 对应的矩阵

$$\boldsymbol{K} := [\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{m\times m} \tag{44}$$

是半正定的, 反之亦然.

证明. 因为核函数可表示为两向量内积: $K_{ij} = \kappa(x_i, x_j) = \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j)$, 令

$$\boldsymbol{\Phi} := [\boldsymbol{\phi}(\boldsymbol{x}_1)\ \boldsymbol{\phi}(\boldsymbol{x}_2)\ \cdots\ \boldsymbol{\phi}(\boldsymbol{x}_m)] \in \mathbb{R}^{\tilde{d}\times m}\,, \tag{45}$$

则 $\boldsymbol{K} = \boldsymbol{\Phi}^\top\boldsymbol{\Phi}$. 对任意非零向量 $\boldsymbol{a}$,

$$\boldsymbol{a}^\top\boldsymbol{K}\boldsymbol{a} = \boldsymbol{a}^\top\boldsymbol{\Phi}^\top\boldsymbol{\Phi}\boldsymbol{a} = (\boldsymbol{\Phi}\boldsymbol{a})^\top(\boldsymbol{\Phi}\boldsymbol{a}) = \|\boldsymbol{\Phi}\boldsymbol{a}\|^2 \geq 0\,. \tag{46}$$

反之亦然.

新的核函数还可以通过现有核函数的组合得到. 使用多个核函数的凸组合是多核学习 [9]的研究内容.

引理 21. 若 $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ 是核函数, 那么下列函数也是核函数.

$$c_1\kappa_1(\boldsymbol{x}_i, \boldsymbol{x}_j) + c_2\kappa_2(\boldsymbol{x}_i, \boldsymbol{x}_j), \quad c_1, c_2 > 0\,, \tag{47}$$

$$\kappa_1(\boldsymbol{x}_i, \boldsymbol{x}_j)\kappa_2(\boldsymbol{x}_i, \boldsymbol{x}_j)\,, \tag{48}$$

$$f(\boldsymbol{x}_1)\kappa_1(\boldsymbol{x}_i, \boldsymbol{x}_j)f(\boldsymbol{x}_2)\,. \tag{49}$$

证明. 因为核函数可表示为两向量内积: $\kappa(x_i, x_j) = \boldsymbol{\phi}(\boldsymbol{x}_i)^\top\boldsymbol{\phi}(\boldsymbol{x}_j)$,

$$c_1\kappa_1(\boldsymbol{x}_i, \boldsymbol{x}_j) + c_2\kappa_2(\boldsymbol{x}_i, \boldsymbol{x}_j) = \begin{bmatrix}\sqrt{c_1}\boldsymbol{\phi}_1(\boldsymbol{x}_i)\\ \sqrt{c_2}\boldsymbol{\phi}_2(\boldsymbol{x}_i)\end{bmatrix}^\top \begin{bmatrix}\sqrt{c_1}\boldsymbol{\phi}_1(\boldsymbol{x}_i)\\ \sqrt{c_2}\boldsymbol{\phi}_2(\boldsymbol{x}_i)\end{bmatrix}, \tag{50}$$

$$\begin{aligned}&\kappa_1(\boldsymbol{x}_i, \boldsymbol{x}_j)\kappa_2(\boldsymbol{x}_i, \boldsymbol{x}_j)\\ &= \operatorname{vec}(\boldsymbol{\phi}_1(\boldsymbol{x}_i)\boldsymbol{\phi}_2(\boldsymbol{x}_i)^\top)^\top \operatorname{vec}(\boldsymbol{\phi}_1(\boldsymbol{x}_j)\boldsymbol{\phi}_2(\boldsymbol{x}_j)^\top)\,,\end{aligned} \tag{51}$$

$$f(\boldsymbol{x}_1)\kappa_1(\boldsymbol{x}_i, \boldsymbol{x}_j)f(\boldsymbol{x}_2) = (f(\boldsymbol{x}_i)\boldsymbol{\phi}(\boldsymbol{x}_i))^\top(f(\boldsymbol{x}_j)\boldsymbol{\phi}(\boldsymbol{x}_j))\,. \tag{52}$$

4.4 核方法

上述核技巧不仅使用于支持向量机, 还适用于一大类问题.

定理 22 (简化版表示定理)**.** 优化问题

$$\min_{\boldsymbol{w}} \quad \frac{1}{m}\sum_{i=1}^m \ell(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i), y_i) + \frac{\lambda}{2}\|\boldsymbol{w}\|^2 \tag{53}$$

的解 $\boldsymbol{w}$ 是样本的线性组合

$$\boldsymbol{w} = \sum_{i=1}^m \alpha_i\boldsymbol{\phi}(\boldsymbol{x}_i)\,. \tag{54}$$

证明. 我们使用反证法. 令

$$\boldsymbol{\Phi} := [\boldsymbol{\phi}(\boldsymbol{x}_1)\ \boldsymbol{\phi}(\boldsymbol{x}_2)\ \cdots\ \boldsymbol{\phi}(\boldsymbol{x}_m)]\,. \tag{55}$$

假设最优解 $\boldsymbol{w}$ 不是样本的线性组合, 那么

$$\exists\boldsymbol{\alpha}, \boldsymbol{e} \neq \boldsymbol{0}.\ \boldsymbol{w} = \boldsymbol{\Phi}\boldsymbol{\alpha} + \boldsymbol{e}\,, \tag{56}$$

其中, $\boldsymbol{e}$ 不是样本的线性组合, 即对任意 $\boldsymbol{\phi}(\boldsymbol{x}_i)$, $\boldsymbol{\phi}(\boldsymbol{x}_i)^\top\boldsymbol{e} = 0$. 因为

$$\begin{aligned}\ell(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i), y_i) &= \ell((\boldsymbol{\Phi}\boldsymbol{\alpha}+\boldsymbol{e})^\top\boldsymbol{\phi}(\boldsymbol{x}_i), y_i)\\ &= \ell((\boldsymbol{\Phi}\boldsymbol{\alpha})^\top\boldsymbol{\phi}(\boldsymbol{x}_i), y_i)\,;\end{aligned} \tag{57}$$

$$\|\boldsymbol{w}\|^2 = \|\boldsymbol{\Phi}\boldsymbol{\alpha}\|^2 + \|\boldsymbol{e}\|^2 + 2(\boldsymbol{\Phi}\boldsymbol{\alpha})^\top\boldsymbol{e} > \|\boldsymbol{\Phi}\boldsymbol{\alpha}\|^2\,, \tag{58}$$

即 $\boldsymbol{\Phi}\boldsymbol{\alpha}$ 比 $\boldsymbol{w}$ 有更小的目标函数值, 说明 $\boldsymbol{w}$ 不是最优解, 与假设矛盾. 因此, 最优解必定是样本的线性组合.

此外, 原版表示定理适用于任意单调递增正则项 $\Omega(\boldsymbol{w})$. 此证明已超出本文范围, 感兴趣的读者可参考 [13].

表示定理对损失函数形式没有限制, 这意味着对许多优化问题, 最优解都可以写成样本的线性组合. 更进一步, $\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x})$ 将可以写成核函数的线性组合

$$\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}) = \sum_{i=1}^m \alpha_i\kappa(\boldsymbol{x}_i, \boldsymbol{x})\,. \tag{59}$$

通过核函数, 我们可以将线性模型扩展成非线性模型. 这启发了一系列基于核函数的学习方法, 统称为核方法 [8].

5 软间隔

不管直接在原特征空间, 还是在映射的高维空间, 我们都假设样本是线性可分的. 虽然理论上我们总能找到一个高维映射使数据线性可分, 但在实际任务中, 寻找到这样一个合适的核函数通常很难. 此外, 由于数据中通常有噪声存在, 一味追求数据线性可分可能会使模型陷入过拟合的泥沼. 因此, 我们放宽对样本的要求, 即允许有少量样本分类错误.

Table 1: 常用核函数. 除此之外, 还有其他一些核函数, 例如卡方核 (chi squared kernel), 直方图交叉核 (histogram intersection kernel) 等.

名称	形式	优点	缺点
线性核	$\boldsymbol{x}_i^\top \boldsymbol{x}_j$	有高效实现, 不易过拟合	无法解决非线性可分问题
多项式核	$(\beta \boldsymbol{x}_i^\top \boldsymbol{x}_j + \theta)^n$	比线性核更一般, n 直接描述了被映射空间的复杂度	参数多, 当 n 很大时会导致计算不稳定
RBF 核	$\exp\left(-\frac{\lVert \boldsymbol{x}_i - \boldsymbol{x}_j \rVert^2}{2\sigma^2}\right)$	只有一个参数, 没有计算不稳定问题	计算慢, 过拟合风险大

5.1 软间隔支持向量机基本型

我们希望在优化间隔的同时, 允许分类错误的样本出现, 但这类样本应尽可能少:

$$\begin{aligned} \min_{\boldsymbol{w}, b} \quad & \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{w} + C \sum_{i=1}^{m} \mathbb{I}(y_i \neq \operatorname{sign}(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b)) \qquad (60) \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b) \geq 1, \quad \text{若 } y_i = \operatorname{sign}(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b)\,. \end{aligned}$$

其中, $\mathbb{I}(\cdot)$ 是指示函数, C 是个可调节参数用于权衡优化间隔和少量分类错误样本这两个目标. 但是, 指示函数不连续, 更不是凸函数, 使得优化问题不再是二次规划问题. 所以我们需要对其进行简化.

公式 60难以实际应用的原因在于指示函数只有两个离散取值 0/1, 对应样本分类正确/错误. 为了能使优化问题继续保持为二次规划问题, 我们需要引入一个取值为连续值的变量, 刻画样本满足约束的程度. 我们引入松弛变量 (slack variable) ξ_i, 用于度量样本违背约束的程度. 当样本违背约束的程度越大, 松弛变量值越大. 即,

$$\xi_i = \begin{cases} 0 & \text{若 } y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b) \geq 1\,; \\ 1 - y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b) & \text{否侧}\,. \end{cases} \tag{61}$$

定理 23 (软间隔支持向量机基本型)**.** *软间隔支持向量机旨在找到一组合适的参数 $(\boldsymbol{w}, b)$, 使得*

$$\begin{aligned} \min_{\boldsymbol{w}, b, \boldsymbol{\xi}} \quad & \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{w} + C \sum_{i=1}^{m} \xi_i \qquad (62) \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b) \geq 1 - \xi_i, \quad i = 1, 2, \ldots, m\,, \\ & \xi_i \geq 0, \quad i = 1, 2, \ldots, m\,. \end{aligned}$$

其中, C 是个可调节参数用于权衡优化间隔和少量样本违背大间隔约束这两个目标. 当 C 比较大时, 我们希望更多的样本满足大间隔约束; 当 C 比较小时, 我们允许有一些样本不满足大间隔约束.

证明. 当样本满足约束 $y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b) \geq 1$ 时, $y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b) \geq 1 - \xi_i$ 对任意 $\xi_i \geq 0$ 成立, 而优化目标要最小化 ξ_i, 所以 $\xi_i = 0$. 当样本不满足约束时, $\xi_i \geq 1 - y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b)$, 而优化目标要最小化 ξ_i, 所以 $\xi_i = 1 - y_i(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b)$.

推论 24. *软间隔支持向量机基本型中描述的优化问题属于二次规划问题, 包括 $m + \tilde{d} + 1$ 个优化变量, $2m$ 项约束.*

证明. 令

$$\boldsymbol{u} := \begin{bmatrix} \boldsymbol{w} \\ b \\ \boldsymbol{\xi} \end{bmatrix}, \ \boldsymbol{Q} := \begin{bmatrix} \boldsymbol{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \ \boldsymbol{t} := C \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix}, \tag{63}$$

$$\boldsymbol{c}_i := \begin{bmatrix} y_i \boldsymbol{\phi}(x_i) \\ y_i \\ \mathbf{e}_i \end{bmatrix}, \ d_i := 1, \quad i = 1, 2, \ldots, m\,, \tag{64}$$

$$\boldsymbol{c}_i := \begin{bmatrix} \mathbf{0} \\ \mathbf{e}_i \end{bmatrix}, \ d_i := 0, \quad i = m+1, \ldots, 2m\,, \tag{65}$$

代入公式 10即得.

5.2 软间隔支持向量机对偶型

定理 25 (软间隔支持向量机对偶型)**.** *软间隔支持向量机的对偶问题等价于找到一组合适的 $\boldsymbol{\alpha}$, 使得*

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j) - \sum_{i=1}^{m} \alpha_i \qquad (66) \\ \text{s.t.} \quad & \sum_{i=1}^{m} \alpha_i y_i = 0, \end{aligned}$$

$$0 \le \alpha_i \le \xi_i, \quad i = 1, 2, \ldots, m\,.$$

证明. 软间隔支持向量机的拉格朗日函数为

$$\begin{aligned}\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) := \frac{1}{2}\boldsymbol{w}^\top\boldsymbol{w} + C\sum_{i=1}^m \xi_i \\ + \sum_{i=1}^m \alpha_i(1 - \xi_i - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b)) \\ + \sum_{i=1}^m \beta_i(-\xi_i)\,. \end{aligned} \tag{67}$$

其对偶问题为

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad \mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \tag{68}$$

$$\text{s.t.} \quad \alpha_i \ge 0, \quad i = 1, 2, \ldots, m\,, \tag{69}$$

$$\beta_i \ge 0, \quad i = 1, 2, \ldots, m\,.$$

因为内层对 $(\boldsymbol{w}, b, \boldsymbol{\xi})$ 的优化属于无约束优化问题, 我们可以通过令偏导等于零的方法得到 $(\boldsymbol{w}, b, \boldsymbol{\xi})$ 的最优值.

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = \mathbf{0} \Rightarrow \boldsymbol{w} = \sum_{i=1}^m \alpha_i y_i \boldsymbol{\phi}(\boldsymbol{x}_i)\,, \tag{70}$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \Rightarrow \sum_{i=1}^m \alpha_i y_i = 0\,, \tag{71}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\xi}} = \mathbf{0} \Rightarrow \alpha_i + \beta_i = C\,. \tag{72}$$

因为存在约束 $\beta_i = C - \alpha_i \ge 0$, 不失一般性, 我们可以约束 $0 \le \alpha_i \le C$, 从而去掉变量 β_i. 将其代入公式 68, 消去 $(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\beta})$, 即得.

推论 26. *软间隔支持向量机对偶型中描述的优化问题属于二次规划问题, 包括 m 个优化变量, $2m+2$ 项约束.*

证明. 令

$$\boldsymbol{u} := \boldsymbol{\alpha},\ \boldsymbol{Q} := [y_i y_j \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j)]_{m\times m},\ \boldsymbol{t} := -\mathbf{1}\,, \tag{73}$$

$$\boldsymbol{c}_i := \mathbf{e}_i,\ d_i := 0, \quad i = 1, 2, \ldots, m\,, \tag{74}$$

$$\boldsymbol{c}_i := -\mathbf{e}_i,\ d_i := -\xi_i,\ i = m+1, \ldots, 2m\,, \tag{75}$$

$$\boldsymbol{c}_{2m+1} := [y_1\ y_2\ \cdots\ y_m]^\top,\ d_{2m+1} := 0\,, \tag{76}$$

$$\boldsymbol{c}_{2m+2} := -[y_1\ y_2\ \cdots\ y_m]^\top,\ d_{2m+2} := 0\,, \tag{77}$$

代入公式 10即得.

5.3 软间隔支持向量机的支持向量

定理 27 (软间隔支持向量机的 KKT 条件)**.** *软间隔支持向量机的 KKT 条件如下.*

- *主问题可行:* $1 - \xi_i - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b) \le 0$, $-\xi_i \le 0$;
- *对偶问题可行:* $\alpha_i \ge 0$, $\beta_i \ge 0$;
- *互补松弛:* $\alpha_i(1 - \xi_i - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b)) = 0$, $\beta_i\xi_i = 0$.

证明. 令

$$\boldsymbol{u} := \begin{bmatrix} \boldsymbol{w} \\ b \\ \boldsymbol{\xi} \end{bmatrix}, \tag{78}$$

$$g_i(\boldsymbol{u}) := 1 - \begin{bmatrix} y_i\boldsymbol{w} \\ y_i \\ \mathbf{e}_i \end{bmatrix}^\top \boldsymbol{u}, \quad i = 1, 2, \ldots, m\,, \tag{79}$$

$$g_i(\boldsymbol{u}) := -\begin{bmatrix} \mathbf{0} \\ \mathbf{e}_i \end{bmatrix}^\top \boldsymbol{u}, \quad i = m+1, \ldots, 2m\,. \tag{80}$$

代入引理 8即得.

引理 28. *软间隔支持向量机中, 支持向量落在最大间隔边界, 内部, 或被错误分类的样本.*

证明. 由软间隔支持向量机的 KKT 条件可知, $\alpha_i(1 - \xi_i - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b)) = 0$ 且 $\beta_i\xi_i = 0$. 当 $\alpha_i > 0$ 时, $1 - \xi_i - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b) = 0$. 进一步可分为两种情况.

- $0 < \alpha_i < C$. 此时 $\beta_i = C - \alpha_i > 0$. 因此 $\xi_i = 0$, 即该样本恰好落在最大间隔边界上;
- $\alpha_i = C$. 此时 $\beta_i = C - \alpha_i = 0$. 若 $\xi_i \le 1$, 该样本落在最大间隔内部; 若 $\xi_i > 1$, 该样本被错误分类.

定理 29. *支持向量机的参数 $(\boldsymbol{w}, b)$ 仅由支持向量决定, 与其他样本无关.*

证明. 和线性支持向量机证明方式相同.

5.4 铰链损失

引理 30. *公式 61等价为*

$$\xi_i = \max(0, 1 - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b))\,. \tag{81}$$

证明. 当样本满足约束时, $1 - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b) \le 0$, $\xi_i = 0$; 当样本不满足约束时, $1 - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b) > 0$, $\xi_i = 1 - y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + b)$.

定理 31. 软间隔支持向量机的基本型等价于

$$\min_{\boldsymbol{w},b} \quad \frac{1}{m}\sum_{i=1}^{m}\max(0, 1-y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i)+b)) + \frac{\lambda}{2}\|\boldsymbol{w}\|^2 \,. \tag{82}$$

其中，第一项称为经验风险，度量了模型对训练数据的拟合程度；第二项称为结构风险，也称为正则化项，度量了模型自身的复杂度．正则化项削减了假设空间，从而降低过拟合风险．λ 是个可调节的超参数，用于权衡经验风险和结构风险．

证明．对应于软间隔支持向量机的基本型，$\xi_i = \max(0, 1-y_i(\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x}_i)+b)) \geq 0$，且 $\lambda = \frac{m}{C}$．

定义 6 (铰链损失 (hinge loss))**.** 铰链损失函数定义为

$$\ell(s) = \max(0, 1-s)\,. \tag{83}$$

除铰链损失外，还有其他一些常用损失函数 [19]，见表 2．$s := y_i\boldsymbol{w}^\top\boldsymbol{\phi}(\boldsymbol{x})$ 的数值大小度量了模型认为该样本属于某一标记的确信程度．我们希望，当样本分类正确时，即 $s > 0$ 时，$\ell(s)$ 小一些；当样本分类错误时，即 $s < 0$ 时，$\ell(s)$ 大一些．

6 优化方法

6.1 SMO

如果直接用经典的二次规划软件包求解支持向量机对偶型，由于 $\boldsymbol{Q} := [y_iy_j\boldsymbol{\phi}(\boldsymbol{x}_i)^\top\boldsymbol{\phi}(\boldsymbol{x}_j)]_{m\times m}$ 的存储开销是 $\mathcal{O}(m^2)$，当训练样本很多时，这将是一个很大的存储和计算开销．序列最小化 (SMO) [10]是一个利用支持向量机自身特性高效的优化算法．SMO 的基本思路是坐标下降．

定义 7 (坐标下降)**.** 通过循环使用不同坐标方向，每次固定其他元素，只沿一个坐标方向进行优化，以达到目标函数的局部最小，见算法 1．

我们希望在支持向量机中的对偶型中，每次固定除 α_i 外的其他变量，之后求在 α_i 方向上的极值．但由于约束 $\sum_{i=1}^m y_i\alpha_i = 0$，当其他变量固定时，$\alpha_i$ 也随着确定．这样，我们无法在不违背约束的前提下对 α_i 进行优化．因此，SMO 每步同时选择两个变量 α_i 和 α_j 进行优化，并固定其他参数，以保证不违背约束．

Algorithm 1 坐标下降.

Input: 优化目标 f.
Output: $\boldsymbol{u}$，使得 $f(\boldsymbol{u})$ 最小.

```
while 不收敛 do
    for i ← 1 to n do
        u_i ← arg min_{u_i} f(u)
    end for
end while
return u
```

定理 32 (SMO 每步的优化目标)**.** *SMO 每步的优化目标为*

$$\begin{aligned}
\min_{\alpha_i,\alpha_j} \quad & \frac{1}{2}(\alpha_i^2y_i^2\boldsymbol{\phi}(\boldsymbol{x}_i)^\top\boldsymbol{\phi}(\boldsymbol{x}_i) + \alpha_i^2y_i^2\boldsymbol{\phi}(\boldsymbol{x}_j)^\top\boldsymbol{\phi}(\boldsymbol{x}_j) \\
& + 2\alpha_i\alpha_jy_iy_j\boldsymbol{\phi}(\boldsymbol{x}_j)^\top\boldsymbol{\phi}(\boldsymbol{x}_j)) - (\alpha_i+\alpha_j) \qquad (84)\\
\mathrm{s.t.} \quad & \alpha_iy_i + \alpha_jy_j = c, \\
& 0 \leq \alpha_i \leq \xi_i, \\
& 0 \leq \alpha_j \leq \xi_j\,,
\end{aligned}$$

其中，$c := -\sum_{k\neq i,j}\alpha_ky_k$．

证明．固定住公式 68中取 α_i，α_j 外的其他变量即得．

推论 33. *SMO 每步的优化目标可等价为对 α_i 的单变量二次规划问题．*

证明．由于 $\alpha_j = y_j(c-\alpha_iy_i)$，我们可以将其代入 SMO 每步的优化目标，以消去变量 α_j．此时，优化目标函数是对于 α_i 的二次函数，约束是一个取值区间 $L \leq \alpha_i \leq H$．之后根据目标函数顶点与区间 $[L, H]$ 的位置关系，可以得到 α_i 的最优值．

理论上讲，每步优化时 α_i 和 α_j 可以任意选择，但实践中通常取 α_i 为违背 KKT 条件最大的变量，而 α_j 取对应样本与 α_i 对应样本之间间隔最大的变量．对 SMO 算法收敛性的测试可以用过检测是否满足 KKT 条件得到．

6.2 Pegasos

我们也可以直接在原问题对支持向量机进行优化，尤其是使用线性核函数时，我们有很高效的优化算法，如 Pegasos [14]. Pegasos 使用基于梯度的方法在线性支持向量机基本型

$$\min_{\boldsymbol{w},b} \quad \frac{1}{m}\sum_{i=1}^{m}\max(0, 1-y_i(\boldsymbol{w}^\top\boldsymbol{x}_i+b)) + \frac{\lambda}{2}\|\boldsymbol{w}\|^2 \,. \tag{85}$$

Table 2: 常用损失函数. 其中 $s := y_i \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x})$.

名称	形式	特点	实例
0/1 损失	$\mathbb{I}(s<0)$	直接优化目标; 非凸, 不连续, NP 难	感知机
铰链损失	$\max(0, 1-s)$	替代损失, 0/1 损失上界; 凸, 连续	支持向量机, 基于二次规划方法优化
对数几率损失	$\log(1+\exp(-s))$	替代损失, 0/1 损失上界; 凸, 连续	对数几率回归, 基于梯度下降方法优化
指数损失	$\exp(-s)$	替代损失, 0/1 损失上界; 凸, 连续	AdaBoost, 分布优化基分类器权重

进行优化, 见算法 2.

Algorithm 2 Pegasos.

Input: $\{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$.

Output: 支持向量机参数 $(\boldsymbol{w}, b)$

```
1: while 不收敛 do
2:     ∂J/∂w ← −(1/m) Σ_{i=1}^m 𝕀(y_i(wᵀx_i + b) ≤ 1) · y_i x_i + λw
3:     ∂J/∂w ← −(1/m) Σ_{i=1}^m 𝕀(y_i(wᵀx_i + b) ≤ 1) · y_i
4:     w ← w − η ∂J/∂w
5:     b ← b − η ∂J/∂b
6: end while
7: return (w, b)
```

6.3 近似算法

当使用非线性核函数下的支持向量机时, 由于核矩阵 $\boldsymbol{K} := [\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)]_{m\times m}$, 所以时间复杂度一定是 $\Omega(m^2)$. 因此, 有许多学者致力于研究一些快速的近似算法. 例如, CVM [15]基于近似最小包围球算法, Nyström 方法 [18]通过从 $\boldsymbol{K}$ 采样出一些列来得到 $\boldsymbol{K}$ 的低秩近似, 随机傅里叶特征 [12]构造了向低维空间的随机映射.

本章介绍了许多优化算法, 实际上现在已有许多开源软件包对这些算法有很好的实现, 目前比较著名的有 LibLinear [7] 和 LibSVM [3], 分别适用于线性和非线性核函数.

7 支持向量机的其他变体

ProbSVM. 对数几率回归可以估计出样本属于正类的概率, 而支持向量机只能判断样本属于正类或负类, 无法得到概率. ProbSVM [11]先训练一个支持向量机, 得到参数 $(\boldsymbol{w}, b)$. 再令 $s_i := y_i \boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b$, 将 $\{(s_1, y_1), (s_2, y_2), \ldots, (s_m, y_m)\}$ 当做新的训练数据训练一个对数几率回归模型, 得到参数 (θ_1, θ_0). 因此, ProbSVM 的假设函数为

$$h(\boldsymbol{x}) := \operatorname{sigm}(\theta_1(\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}) + b) + \theta_0) \,. \tag{86}$$

对数几率回归模型可以认为是对训练得到的支持向量机的微调, 包括尺度 (对应 θ_1) 和平移 (对应 θ_0). 通常 $\theta_1 > 0, \theta_0 \approx 0$.

多分类支持向量机. 支持向量机也可以扩展到多分类问题中. 对于 K 分类问题, 多分类支持向量机 [17] 有 K 组参数 $\{(\boldsymbol{w}_1, b_1), (\boldsymbol{w}_2, b_2), \ldots, (\boldsymbol{w}_K, b_K)\}$, 并希望模型对于属于正确标记的结果以 1 的间隔高于其他类的结果, 形式化如下

$$\begin{aligned}\min_{\boldsymbol{W}, \boldsymbol{b}} \quad & \frac{1}{m}\sum_{i=1}^m \sum_{k=1}^K \max(0, (\boldsymbol{w}_{y_i}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b_{y_i}) \\ & -(\boldsymbol{w}_k^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b_k) + 1) + \frac{\lambda}{2}\sum_{k=1}^K \boldsymbol{w}_k^\top \boldsymbol{w}_k \,.\end{aligned} \tag{87}$$

支持向量回归 (SVR). 经典回归模型的损失函数度量了模型的预测 $h(\boldsymbol{x}_i)$ 和 y_i 的差别, 支持向量回归 [6]能够容忍 $h(\boldsymbol{x}_i)$ 与 y_i 之间小于 ε 的偏差. 令 $s := y - (\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}) + b$, 我们定义 ε-不敏感损失为

$$\begin{cases} 0 & \text{若 } |y - (\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}) + b)| \le \varepsilon\,; \\ |s| - \varepsilon & \text{否则}\,. \end{cases} \tag{88}$$

定理 34 (支持向量回归). *支持向量回归可形式化为*

$$\min_{\boldsymbol{w}, b} \quad \frac{1}{m}\sum_{i=1}^m \max(0, |y_i - (\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}_i) + b)| - \varepsilon) + \frac{\lambda}{2}\boldsymbol{w}^\top \boldsymbol{w} \,. \tag{89}$$

证明. 当 $|y - (\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}) + b)| \le \varepsilon$ 时, $|y - (\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}) + b)| - \varepsilon \le 0$, $\max(0, \cdot)$ 结果为 0; 当 $|y - (\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}) + b)| > \varepsilon$ 时, $\max(0, \cdot)$ 结果为 $|y - (\boldsymbol{w}^\top \boldsymbol{\phi}(\boldsymbol{x}) + b)| - \varepsilon$.

References

[1] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Annual Workshop on Computational Learning Theory*, pages 144–152, 1992. 5

[2] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004. 4

[3] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27, 2011. 10

[4] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995. 1

[5] N. Cristianini and J. Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, 2000. 6

[6] H. Drucker, C. J. Burges, L. Kaufman, A. J. Smola, and V. Vapnik. Support vector regression machines. In *Advances in Neural Information Processing Systems*, pages 155–161, 1997. 10

[7] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9(8):1871–1874, 2008. 10

[8] T. Hofmann, B. Schölkopf, and A. J. Smola. Kernel methods in machine learning. *The Annals of Statistics*, pages 1171–1220, 2008. 6

[9] G. R. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5(1):27–72, 2004. 6

[10] J. Platt. Sequential minimal optimization: A fast algorithm for training support vector machines. *Micriosoft Research*, 1998. 9

[11] J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3):61–74, 1999. 10

[12] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, pages 1177–1184, 2008. 10

[13] B. Scholkopf and A. J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2001. 6

[14] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter. Pegasos: Primal estimated sub-gradient solver for SVM. *Mathematical Programming*, 127(1):3–30, 2011. 9

[15] I. W. Tsang, J. T. Kwok, and P.-M. Cheung. Core vector machines: Fast SVM training on very large data sets. *Journal of Machine Learning Research*, 6(4):363–392, 2005. 10

[16] V. Vapnik. *The nature of statistical learning theory*. Springer Science & Business Media, 2013. 5

[17] J. Weston, C. Watkins, et al. Support vector machines for multi-class pattern recognition. In *Proceedings of the European Symposium on Artificial Neural Networks*, volume 99, pages 219–224, 1999. 10

[18] C. K. Williams and M. Seeger. Using the nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems*, pages 682–688, 2001. 10

[19] 周志华. 机器学习. 清华大学出版社, 2016. 9