凸优化与支持向量机

王璐

对于没有限制条件的凸优化，如果目标函数可导，则使目标函数梯度为 $\mathbf{0}$ 的点是全局最小值点。本章以支持向量机 (Support Vector Machine, 简称 SVM) 为例，介绍带限制条件的凸优化问题的一般解法。SVM 是最好的监督学习 (supervised learning) 算法之一。监督学习就是从一些事先标记过的训练数据中建立一个模型或学习一个函数，这个模型或函数可以对输入的特征做出预测（输出）。

1 SVM: Margins

本节将介绍 SVM 的一个重要概念 – margin. Margin 代表一种预测的 "信心"。在 logistic 回归中，我们用 logistic 函数预测概率

$$P(Y=1 \mid \boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{1+\exp(-\boldsymbol{\theta}^\top \boldsymbol{x})}.$$

决策时可以采用如下规则：如果 $P(Y=1 \mid \boldsymbol{x}, \boldsymbol{\theta}) \geq 0.5$, 预测 $Y=1$, 反之 $Y=0$. 或者等价地，如果 $\boldsymbol{\theta}^\top \boldsymbol{x} \geq 0$, 预测 $Y=1$, 反之 $Y=0$. $\boldsymbol{\theta}^\top \boldsymbol{x}$ 的值越大，$P(Y=1 \mid \boldsymbol{x}, \boldsymbol{\theta})$ 越接近 1，我们对预测 $Y=1$ 越有信心; 同理如果 $P(Y=1 \mid \boldsymbol{x}, \boldsymbol{\theta}) \approx 0$, 我们对预测 $Y=0$ 就越有信心。图1展示了一个数据集，其中 $\times$ 代表标记为 1 的点，$\circ$ 代表标记为 0 的点。图1中的实线是用这些训练数据得到的一条决策边界 (decision boundary): $\boldsymbol{\theta}^\top \boldsymbol{x} = 0$. 图1中的 A, B, C 是要预测的点，其中 A 点远离决策边界且 $\boldsymbol{\theta}^\top \boldsymbol{x}_A \gg 0$, 因此我们对预测 A 点值为 1 很有信心; 相反，C 点很靠近边界，尽管依据决策规则 ($\boldsymbol{\theta}^\top \boldsymbol{x}_C > 0$) 预测 C 点值为 1, 但只要稍微变动一下决策边界，C 点的预测可能就变为 0. 因此我们对 C 的预测没有对 A 的预测那么有信心，对 B 预测的信心介于两者之间。图1表明：当要预测的点越远离决策边界, 我们对它的预测越有信心。

SVM 既可以预测分类，也可以预测连续值，以下我们先从一个最简单的线性 SVM 分类器入手引入 margin 的概念。在训练集 $S = \{(\boldsymbol{x}_i, y_i) : i = 1, \ldots, n\}$ 中，每个点 i 由一个特征 (feature) 向量 $\boldsymbol{x}_i$ 和一个标签 y_i 组成，为了后续计算方便，令 $y_i \in \{-1, 1\}$ (注意不是 $\{0, 1\}$). 我们希望决策边界具有如下形式：

$$\boldsymbol{\omega}^\top \boldsymbol{x} + b = 0. \tag{1}$$

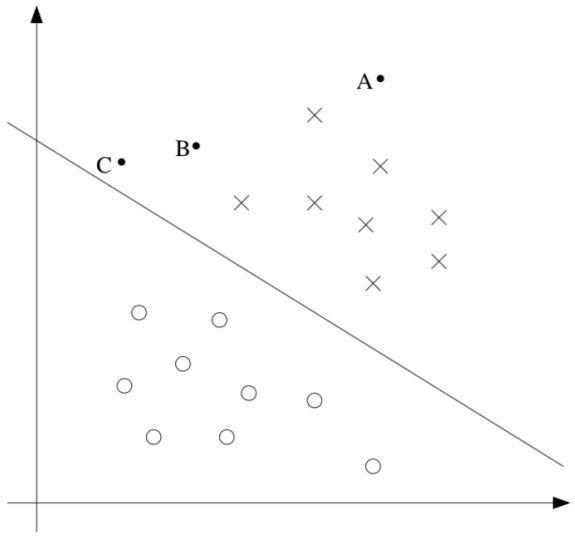

Figure 1: 对 A, B, C 三点的预测信心。Picture source: Andrew Ng

由于 SVM 对截距项的计算与其它系数不同，所以在(1)中将截距项 b 单独写出来。有了决策边界，决策规则为：如果 $\boldsymbol{\omega}^\top \boldsymbol{x} + b \geq 0$, 预测 $y = 1$; 反之，预测 $y = -1$.

如果将点 i 的 **margin** γ_i 定义为

$$\gamma_i = y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) \tag{2}$$

可以看到 $\gamma_i > 0$ 表明对点 i 的预测是正确的，同时较大的 γ_i 代表对预测值较大的信心。但是(2)有一个问题：如果将 $\boldsymbol{\omega}$ 和 b 同时扩大 2 倍，决策边界不变，但是对预测的信心，即 margin γ_i 却扩大了 2 倍。为了保证 margin 可识别，需要对(2)中的系数加一些规范化条件, 比如令 $\|\boldsymbol{\omega}\|_2 = 1$ 或者令

$$\gamma_i = y_i \left(\frac{\boldsymbol{\omega}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{\omega}\|_2} \right). \tag{3}$$

以下将 $\|\cdot\|_2$ 简记为 $\|\cdot\|$.

现在我们来考察一下定义(3)的**几何意义**。在图2中，A 点的坐标为 $\boldsymbol{x}_i$, 同时 $y_i = 1$; A 点在决策边界 $\boldsymbol{\omega}^\top \boldsymbol{x} + b = 0$ 上的投影为 B. 设 AB 的距离为 d_{AB}, 则由 A 点出发，沿着单位向量 $-\boldsymbol{\omega}/\|\boldsymbol{\omega}\|$ 走 d_{AB} 个单位即到达 B 点，所以 B 的坐标为 $\boldsymbol{x}_i - d_{AB}\boldsymbol{\omega}/\|\boldsymbol{\omega}\|$. 注意到 B 点在决策边界上，因此满足

$$\boldsymbol{\omega}^\top \left(\boldsymbol{x}_i - d_{AB} \frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\|} \right) + b = 0.$$

解得

$$d_{AB} = \frac{\boldsymbol{\omega}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{\omega}\|}. \tag{4}$$

比较(3)和(4), 我们证明了 $y_i = 1$ 时，γ_i 等于点 i 到决策边界的距离。类似可证该结论对 $y_i = -1$ 的点也成立。

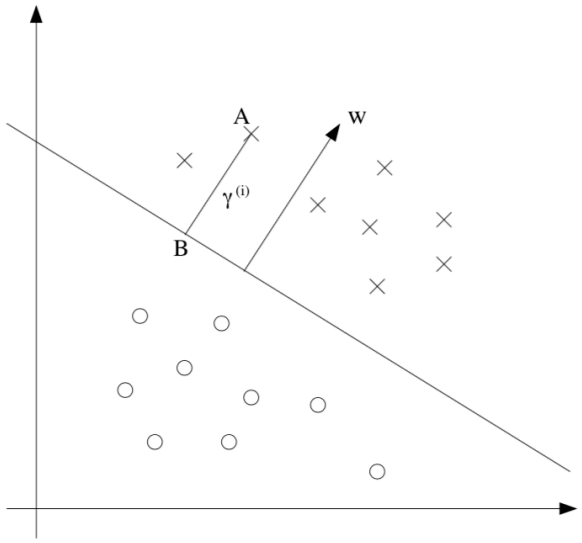

Figure 2: Margin 的几何意义。Picture source: Andrew Ng

假设训练集是线性可分的，即存在超平面 $\boldsymbol{\omega}^\top \boldsymbol{x} + b = 0$ 可以将正负点区分开。我们会在 Section 4 中讨论线性不可分的情形。SVM 希望训练集中的所有点都远离决策边界，令

$$\gamma = \min_i \ \gamma_i$$

SVM 的目标是寻找一条决策边界使得 γ 最大:

$$\max_{\boldsymbol{\omega},b} \ \gamma \ \ \text{s.t.} \ \ y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b)/\left\|\boldsymbol{\omega}\right\| \geq \gamma, \ i = 1, \ldots, n$$

它等价于

$$\max_{\boldsymbol{\omega},b} \ \gamma \ \ \text{s.t.} \ \ y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) \geq \gamma \left\|\boldsymbol{\omega}\right\|, \ i = 1, \ldots, n. \tag{5}$$

在(5)中令 $\|\boldsymbol{\omega}\| = 1/\gamma$, 则最大化 γ 等价于最小化 $\|\boldsymbol{\omega}\|$:

$$\min_{\boldsymbol{\omega},b} \ \left\|\boldsymbol{\omega}\right\| \ \ \text{s.t.} \ \ y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) \geq 1, \ i = 1, \ldots, n. \tag{6}$$

为了计算方便，将(6)写为以下形式:

$$\min_{\boldsymbol{\omega},b} \ \frac{1}{2}\left\|\boldsymbol{\omega}\right\|^2 \ \ \text{s.t.} \ \ -y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) + 1 \leq 0, \ i = 1, \ldots, n. \tag{7}$$

此时我们将最大化最小 margin 的问题(5)转化为非常容易求解的优化问题(7). 当优化问题的目标函数和限制条件都是线性函数时，有通用的 linear programming 算法求解；当目标函数是二次函

数、限制条件是线性时，有通用的 quadratic programming (QP) 算法。因此(7)可以用 QP 算法求解。然而在很多实际问题中，特征 $\boldsymbol{x}_i \in \mathbb{R}^d$ 是一个高维向量, 即 $d \gg n$, 此时如果将(7)转化为它的 Lagrange dual form 求解, 会比直接使用 QP 更高效，因为它将一个 d 维优化问题转化为 n 维优化问题，极大减小计算量。这种转化也使得在更高维的空间寻找非线性 margin 决策曲面变得高效。为此我们需要先了解一些凸优化的理论。

2 Convex Optimization

考虑一般的凸优化问题：

$$\begin{aligned} &\min_{\boldsymbol{x}\in\mathbb{R}^d} f(\boldsymbol{x}) \\ &\text{s.t. } g_i(\boldsymbol{x}) \le 0,\ i = 1,\dots,m \\ &\quad h_j(\boldsymbol{x}) = 0,\ j = 1,\dots,p. \end{aligned} \tag{8}$$

其中函数 $f : \mathbb{R}^d \to \mathbb{R}$ 和 $g_i : \mathbb{R}^d \to \mathbb{R}$, $i = 1,\dots,m$ 都是可导的凸函数，函数 $h_j : \mathbb{R}^d \to \mathbb{R}$, $j = 1,\dots,p$ 都是仿射函数 (affine functions).

回顾凸函数和仿射函数的定义。如果函数 $g : G \to \mathbb{R}$ 满足 G 是一个凸集且对于任意两点 $\boldsymbol{x}_1, \boldsymbol{x}_2 \in G$, $\forall \theta \in [0,1]$ 有下式成立：

$$g\left(\theta \boldsymbol{x}_1 + (1-\theta)\boldsymbol{x}_2\right) \le \theta g(\boldsymbol{x}_1) + (1-\theta) g(\boldsymbol{x}_2)$$

称 g 是一个**凸函数**。

仿射函数 $h : \mathbb{R}^d \to \mathbb{R}$ 的形式为 $h(\boldsymbol{x}) = \boldsymbol{a}^\top \boldsymbol{x} + b$, 其中 $\boldsymbol{a} \in \mathbb{R}^d$, $b \in \mathbb{R}$. 仿射函数既是凸函数又是凹函数。

带限制条件的优化问题(8)可以写为以下等价的无限制优化问题：

$$\min_{\boldsymbol{x}} \Theta_P(\boldsymbol{x}) \triangleq f(\boldsymbol{x}) + \infty \sum_{i=1}^{m} \mathbf{1}\left(g_i(\boldsymbol{x}) > 0\right) + \infty \sum_{j=1}^{p} \mathbf{1}\left(h_j(\boldsymbol{x}) \ne 0\right) \tag{9}$$

称(9)为 **primal optimization**. 但是(9)很难求解因为目标函数 $\Theta_P(\boldsymbol{x})$ 不连续更不可导。考虑用某种可导函数替换惩罚函数 $\infty \cdot \mathbf{1}(u > 0)$，比如线性函数 αu. 由于 $\infty \cdot \mathbf{1}(u > 0)$ 只惩罚 $u > 0$ 的部分，当 $\alpha \ge 0$ 时，函数 αu 是 $\infty \cdot \mathbf{1}(u > 0)$ 的一个下界函数，如图3所示。类似地，函数 βu 总是 $\infty \cdot \mathbf{1}(u \ne 0)$ 的一个下界函数（β 的取值没有限制）。

定义 **Lagrangian**：

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = f(\boldsymbol{x}) + \sum_{i=1}^{m} \alpha_i g_i(\boldsymbol{x}) + \sum_{j=1}^{p} \beta_j h_j(\boldsymbol{x}). \tag{10}$$

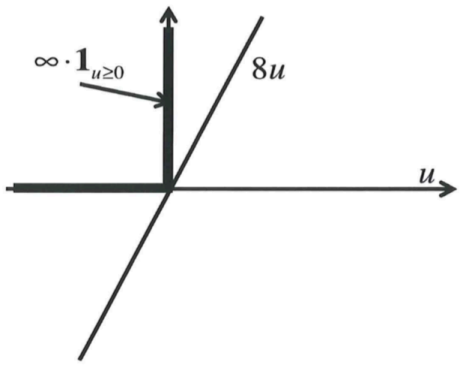

Figure 3: 惩罚函数 $\infty \cdot \mathbf{1}(u \geq 0)$ 和它的一个下界函数 $8u$. Picture source: Cynthia Rudin

称(10)中 $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m)$ 和 $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$ 的元素为 Lagrange multipliers. 可以证明

$$\Theta_P(\boldsymbol{x}) = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \text{ s.t. } \alpha_i \geq 0, \forall i. \tag{11}$$

Proof. 对于任意 $\boldsymbol{x}$,

- 如果某个限制条件成立：假设 $g_i(\boldsymbol{x}) \leq 0$, 为使 $\alpha_i g_i(\boldsymbol{x})$ 尽可能大，应该令 $\alpha_i = 0$; 如果 $h_j(\boldsymbol{x}) = 0$, 则 β_j 取任何值都不会改变 $\mathcal{L}$ 的值。
- 如果某个限制条件不成立：假设 $g_i(\boldsymbol{x}) > 0$, 为使 $\alpha_i g_i(\boldsymbol{x})$ 尽可能大，应该令 $\alpha_i = \infty$; 如果 $h_j(\boldsymbol{x}) \neq 0$, 为使 $\beta_j h_j(\boldsymbol{x})$ 尽可能大 (达到 ∞)，应该令 $\beta_j = +\infty$ 或 $-\infty$.

因此通过调整 α_i's 和 β_j's 的值总可以使(11)成立。 □

由(11)可知 $\Theta_P(\boldsymbol{x})$ 是 $\boldsymbol{x}$ 的凸函数。首先, $f(\boldsymbol{x})$ 是凸函数。其次，每个 $g_i(\boldsymbol{x})$ 是凸函数，$\alpha_i \geq 0$，因此每个 $\alpha_i g_i(\boldsymbol{x})$ 是凸函数。由于每个 $h_j(\boldsymbol{x})$ 是线性函数，不论 β_j 的符号正或负, $\beta_j h_j(\boldsymbol{x})$ 总是凸函数。凸函数的和仍是凸函数，所以 $\mathcal{L}$ 是 $\boldsymbol{x}$ 的凸函数。最后，一列凸函数的上确界仍是凸函数。所以 $\Theta_P(\boldsymbol{x})$ 是 $\boldsymbol{x}$ 的凸函数。

根据(11)，可以将 primal optimization (9)转化为以下目标函数可导的优化问题：

$$\min_{\boldsymbol{x}} \Theta_P(\boldsymbol{x}) = \min_{\boldsymbol{x}} \left[\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\alpha_i \geq 0,\forall i} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \right]. \tag{12}$$

如果点 $\boldsymbol{x}$ 满足所有限制条件，即 $g_i(\boldsymbol{x}) \leq 0, \forall i$ 且 $h_j(\boldsymbol{x}) = 0, \forall j$, 称点 $\boldsymbol{x}$ 为 **primal feasible**. 假设点 $\boldsymbol{x}^\star$ 使 $\Theta_P(\boldsymbol{x})$ 达到最小，最小值为 $p^\star = \Theta_P(\boldsymbol{x}^\star)$.

如果交换(12)中 min 和 max 的顺序，就得到了另一个不同的优化问题，称为(12)的 dual problem：

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\alpha_i\geq 0,\forall i}\left[\min_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\beta})\right]=\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\alpha_i\geq 0,\forall i}\Theta_D(\boldsymbol{\alpha},\boldsymbol{\beta}) \tag{13}$$

此处定义 dual objective $\Theta_D(\boldsymbol{\alpha},\boldsymbol{\beta})=\min_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\beta})$. 如果 $\alpha_i\geq 0$, $i=1,\ldots,m$, 称点 $(\boldsymbol{\alpha},\boldsymbol{\beta})$ 为 **dual feasible**. 假设 $(\boldsymbol{\alpha}^\star,\boldsymbol{\beta}^\star)$ 使 $\Theta_D(\boldsymbol{\alpha},\boldsymbol{\beta})$ 达到最大，最大值为 $d^\star=\Theta_D(\boldsymbol{\alpha}^\star,\boldsymbol{\beta}^\star)$.

Theorem 1. *对任意一对 primal and dual problems* (12)和(13)*, 总有* $d^\star\leq p^\star$.

Proof. 如果点 $(\boldsymbol{\alpha},\boldsymbol{\beta})$ 是 dual feasible, 则以下下界关系成立：

$$\alpha_i g_i(\boldsymbol{x})\leq\infty\cdot\mathbf{1}\left(g_i(\boldsymbol{x})>0\right),\ \forall i$$

$$\beta_j h_j(\boldsymbol{x})\leq\infty\cdot\mathbf{1}\left(h_j(\boldsymbol{x})\neq 0\right),\ \forall j.$$

因此

$$\mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\beta})\leq\Theta_p(\boldsymbol{x}),\ \forall\boldsymbol{x}.$$

两边关于 $\boldsymbol{x}$ 取最小值，不等式依然成立

$$\underbrace{\min_{\boldsymbol{x}}\ \mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\beta})}_{\Theta_D(\boldsymbol{\alpha},\boldsymbol{\beta})}\leq\underbrace{\min_{\boldsymbol{x}}\ \Theta_p(\boldsymbol{x})}_{p^\star}. \tag{14}$$

(14)对所有 dual feasible 的点 $(\boldsymbol{\alpha},\boldsymbol{\beta})$ 都成立，因此

$$d^\star=\max_{\boldsymbol{\alpha},\boldsymbol{\beta},\alpha_i\geq 0,\forall i}\left[\min_{\boldsymbol{x}}\ \mathcal{L}(\boldsymbol{x},\boldsymbol{\alpha},\boldsymbol{\beta})\right]\leq\min_{\boldsymbol{x}}\ \Theta_p(\boldsymbol{x})=p^\star.$$

□

如果 primal and dual problems 满足 $d^\star=p^\star$，称这种情况为 **strong duality**. 很多条件可以保证 strong duality 成立，最常用的是 **Slater's condition**: 如果优化问题(8)的解 $\boldsymbol{x}^\star$ 使所有不等式限制条件都严格成立，即 $g_i(\boldsymbol{x}^\star)<0$, $i=1,\ldots,m$, 称 primal/dual problem pair 满足 Slater's condition.

2.1 KKT 条件

对于带限制的优化问题(8), 找到满足 KKT 条件的解等价于找到全局最优解 (global minimum).

之前我们用 $\boldsymbol{x}^\star$ 表示 primal optimization (12)的最优解，用 $(\boldsymbol{\alpha}^\star,\boldsymbol{\beta}^\star)$ 表示 dual optimization (13)的最优解。当 strong duality 成立时，可得到以下结论。

Lemma 1 (Complementary Slackness)**.** *如果 strong duality 成立，那么* $\alpha_i^\star g_i(\boldsymbol{x}^\star)=0$, $i=1,\ldots,m$.

Proof. 由定义出发可得

$$\begin{aligned} d^\star = \Theta_D(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star) &= \min_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star) \\ &\le \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star) \qquad (15) \\ &\le \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \alpha_i \ge 0, \forall i} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \Theta_p(\boldsymbol{x}^\star) \qquad (16) \\ &= f(\boldsymbol{x}^\star) = p^\star \qquad (17) \end{aligned}$$

其中不等式(15)是因为 $\min_{\boldsymbol{x}}$ 小于任意 $\boldsymbol{x}$ 处的值, 当然包括 $\boldsymbol{x}^\star$; 同理可得不等式(16); 等式(17)成立是因为 primal optimization (12)的最优解一定是 primal feasible, 即满足所有限制条件。

当 strong duality 成立时，$d^\star = p^\star$，因此上式中的所有不等式都可以写为等式。此时有

$$\mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star) = f(\boldsymbol{x}^\star) + \sum_{i=1}^m \alpha_i^\star g_i(\boldsymbol{x}^\star) + \sum_{j=1}^p \beta_j^\star h_j(\boldsymbol{x}^\star) = f(\boldsymbol{x}^\star) \tag{18}$$

所以

$$\sum_{i=1}^m \alpha_i^\star g_i(\boldsymbol{x}^\star) + \sum_{j=1}^p \beta_j^\star h_j(\boldsymbol{x}^\star) = 0$$

由于 $\boldsymbol{x}^\star$ 是 primal feasible，因此 $h_j(\boldsymbol{x}^\star) = 0$, $j = 1, \ldots, p$. 所以

$$\sum_{i=1}^m \alpha_i^\star g_i(\boldsymbol{x}^\star) = 0. \tag{19}$$

注意到

(i) 因为 $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)$ 是 dual feasible, 所以 $\alpha_i^\star \ge 0$, $i = 1, \ldots, m$;

(ii) 因为 $\boldsymbol{x}^\star$ 是 primal feasible, 所以 $g_i(\boldsymbol{x}^\star) \le 0$, $i = 1, \ldots, m$.

由 (i)(ii) 可得 $\alpha_i^\star g_i(\boldsymbol{x}^\star) \le 0$, $i = 1, \ldots, m$，再由(19)得

$$\alpha_i^\star g_i(\boldsymbol{x}^\star) = 0, \ i = 1, \ldots, m.$$

□

Remark

- 由 Lemma 1可得，当 strong duality 成立时，在 primal/dual problem 的最优解 $(\boldsymbol{x}^\star, \boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)$ 处有以下结论成立：
  - 如果某个 $\alpha_i^\star > 0$，则对应的 $g_i(\boldsymbol{x}^\star) = 0$，此时称该限制条件 g_i 为 active constraint 或 binding constraint.

– 如果某个 $g_i(\boldsymbol{x}^\star) < 0$, 则对应的 $\alpha_i^\star = 0$.

- 当 strong duality 成立时，根据 Lemma 1的证明，$\boldsymbol{x}^\star$ 是凸函数 $\mathcal{L}(\boldsymbol{x}, \boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)$ 的最小值点，因此满足梯度为零：

$$\nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star) = \nabla_{\boldsymbol{x}} f(\boldsymbol{x}^\star) + \sum_{i=1}^{m} \alpha_i^\star \nabla_{\boldsymbol{x}} g_i(\boldsymbol{x}^\star) + \sum_{j=1}^{p} \beta_j^\star \nabla_{\boldsymbol{x}} h_j(\boldsymbol{x}^\star) = \mathbf{0}. \tag{20}$$

一般称等式(20)为 **Lagrangian stationarity**. (20)表明在最优解 $\boldsymbol{x}^\star$ 处，目标函数 f 的梯度和限制函数的梯度方向相反，模长相等，如图4所示。图4中的曲线代表 f 的等高线 (contours)，直线代表等式限制条件。$\boldsymbol{x}^\star$ 是 primal optimization (12)的最优解，此时目标函数 f 的梯度和限制函数的梯度方向相反、模长相等。从点 $\boldsymbol{x}^\star$ 出发再沿直线移动 $\boldsymbol{x}$ 也不会得到更小的 $f(\boldsymbol{x})$ 的值，因此图4中的直线一定与 f 的等高线相切。

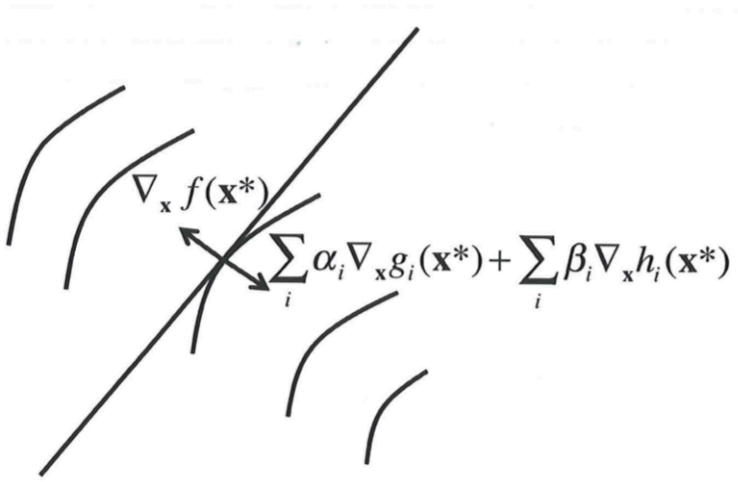

Figure 4: 最优解 $\boldsymbol{x}^\star$ 处目标函数的梯度与限制函数的梯度关系。Picture source: Cynthia Rudin

---

现在可以给出 primal dual optimization pair 的全局最优解满足的条件了，这些条件被称为 Karush-Kuhn-Tucker (KKT) 条件。

Theorem 2 (KKT conditions)**.** *如果点 $\boldsymbol{x}^\star \in \mathbb{R}^d$, $\boldsymbol{\alpha}^\star \in \mathbb{R}^m$, $\boldsymbol{\beta}^\star \in \mathbb{R}^p$ 满足以下条件：*

- *(Primal feasibility)* $g_i(\boldsymbol{x}^\star) \le 0$, $i = 1, \ldots, m$ *且* $h_j(\boldsymbol{x}^\star) = 0$, $j = 1, \ldots, p$.
- *(Dual feasibility)* $\alpha_i^\star \ge 0$, $i = 1, \ldots, m$.
- *(Complementary Slackness)* $\alpha_i^\star g_i(\boldsymbol{x}^\star) = 0$, $i = 1, \ldots, m$.

- *(Lagrangian stationary)* $\nabla_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x}^\star, \boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star) = \mathbf{0}$.

则 $\boldsymbol{x}^\star$ 是 *primal optimal*, $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)$ 是 *dual optimal*. 如果 *strong duality* 成立，则任何 *primal optimal* $\boldsymbol{x}^\star$ 及任何 *dual optimal* $(\boldsymbol{\alpha}^\star, \boldsymbol{\beta}^\star)$ 必须满足以上条件。

Remarks

1. 如果 strong duality 不成立，KKT 条件是找到优化问题(8)全局最优解的充分条件。

2. 如果 strong duality 成立，KKT 条件是找到(8)全局最优解的充要条件。

历史上，KKT 条件最初是 Karush 在硕士论文 (1939) 中提出的，但没有引起任何注意，直到 1950 年两位数学家 Kuhn 和 Tucker 重新发现才获得关注。

3 SVM: Maximize the minimum margin

回到线性 SVM 分类模型，我们要找的最佳决策边界是如下带限制的凸优化问题的解：

$$\begin{aligned} &\min_{\boldsymbol{\omega}, b} \ \frac{1}{2}\|\boldsymbol{\omega}\|^2 \\ &\text{s.t.} \ \ -y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) + 1 \le 0, \ i = 1, \ldots, n. \end{aligned} \tag{21}$$

下面列出(21)的最优解需要满足的 KKT 条件。先从 Lagrangian stationarity 开始，(21)的 Lagrangian 为

$$\begin{aligned} \mathcal{L}([\boldsymbol{\omega}, b], \boldsymbol{\alpha}) &= \frac{1}{2}\sum_{j=1}^{d}\omega_j^2 + \sum_{i=1}^{n}\alpha_i\left[-y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) + 1\right] \\ &= \frac{1}{2}\boldsymbol{\omega}^\top\boldsymbol{\omega} - \boldsymbol{\omega}^\top\left(\sum_{i=1}^{n}\alpha_i y_i \boldsymbol{x}_i\right) - b\left(\sum_{i=1}^{n}\alpha_i y_i\right) + \sum_{i=1}^{n}\alpha_i \end{aligned} \tag{22}$$

找到 $\mathcal{L}$ 关于 $\boldsymbol{\omega}$ 和 b 的梯度并令其等于零：

$$\nabla_{\boldsymbol{\omega}}\mathcal{L} = \boldsymbol{\omega} - \sum_{i=1}^{n}\alpha_i y_i \boldsymbol{x}_i = \mathbf{0} \implies \boldsymbol{\omega}^\star = \sum_{i=1}^{n}\alpha_i^\star y_i \boldsymbol{x}_i \tag{23}$$

$$\nabla_b\mathcal{L} = -\sum_{i=1}^{n}\alpha_i y_i = 0 \implies \sum_{i=1}^{n}\alpha_i^\star y_i = 0. \tag{24}$$

列出剩下的 KKT 条件：

$$\alpha_i^\star \ge 0, \quad \forall i \qquad \text{(dual feasibility)} \tag{25}$$

$$-y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) + 1 \le 0, \ \forall i \qquad \text{(primal feasibility)} \tag{26}$$

$$\alpha_i^\star\left[-y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) + 1\right] = 0, \ \forall i \qquad \text{(complementary slackness)} \tag{27}$$

将(23)和(24)代入(22)，得到 dual objective 在 $\boldsymbol{\alpha}^\star$ 处的值：

$$\begin{aligned}\mathcal{L}([\boldsymbol{\omega}^\star, b^\star], \boldsymbol{\alpha}^\star) &= -\frac{1}{2}\boldsymbol{\omega}^{\star\top}\boldsymbol{\omega}^\star + \sum_{i=1}^n \alpha_i^\star \\ &= -\frac{1}{2}\left(\sum_{i=1}^n \alpha_i^\star y_i \boldsymbol{x}_i\right)^\top \left(\sum_{i=1}^n \alpha_i^\star y_i \boldsymbol{x}_i\right) + \sum_{i=1}^n \alpha_i^\star \\ &= -\frac{1}{2}\sum_{i=1}^n\sum_{k=1}^n \alpha_i^\star \alpha_k^\star y_i y_k \boldsymbol{x}_i^\top \boldsymbol{x}_k + \sum_{i=1}^n \alpha_i^\star \\ &= \Theta_D(\boldsymbol{\alpha}^\star)\end{aligned}$$

考虑到 $\boldsymbol{\alpha}^\star$ 还需满足条件(24)和(25), $\boldsymbol{\alpha}^\star$ 是如下 dual optimization 的解：

$$\begin{aligned}&\max_{\boldsymbol{\alpha}} \Theta_D(\boldsymbol{\alpha}) = \sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i=1}^n\sum_{k=1}^n \alpha_i\alpha_k y_i y_k \boldsymbol{x}_i^\top \boldsymbol{x}_k \\ &\text{s.t. } \alpha_i \geq 0,\ i = 1, \ldots, n \\ &\quad \sum_{i=1}^n \alpha_i y_i = 0.\end{aligned} \tag{28}$$

当特征 $\boldsymbol{x}_i$ 的维数 $d \gg n$ 时，与(21)相比，(28)仅对应一个 n 维凸优化，更容易求解。此时可以使用 quadratic programming 计算(28)，或者使用专门为 SVM 设计的 SMO 算法，该算法会在 Section 4.1中详细介绍。

假设已经解出 $\boldsymbol{\alpha}^\star$，则由(23)可以得到 primal optimal：

$$\boldsymbol{\omega}^\star = \sum_{i=1}^n \alpha_i^\star y_i \boldsymbol{x}_i.$$

但是仍然不知道 $b^\star$ 的取值。注意到还有一些 KKT 条件(26)和(27)没有用到，这引出了 SVM 的一个重要概念。

3.1 支持向量

由 complementary slackness 条件(27)可得:

$$\alpha_i^\star\left[-y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) + 1\right] = 0,\ \forall i \implies \begin{cases} \alpha_i^\star > 0 & \Rightarrow y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) = 1 \\ -y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) + 1 < 0 & \Rightarrow \alpha_i^\star = 0 \end{cases}$$

我们重点关注第一类情况，即 $\alpha_i^\star > 0$ 对应的训练集中的点 $(\boldsymbol{x}_i, y_i)$. 可以看到该点对应的 scaled margin $y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) = 1$, 此时不等式限制条件在点 $(\boldsymbol{x}_i, y_i)$ 处以等式成立 (active constraint). 训练集中这样的点 $(\boldsymbol{x}_i, y_i)$ 被称为**支持向量** (support vectors), 它们是最靠近决策边界的点，如图5所示。支持向量到决策边界的距离 (minimum margin) 为 $\gamma = 1/\|\boldsymbol{\omega}^\star\|$.

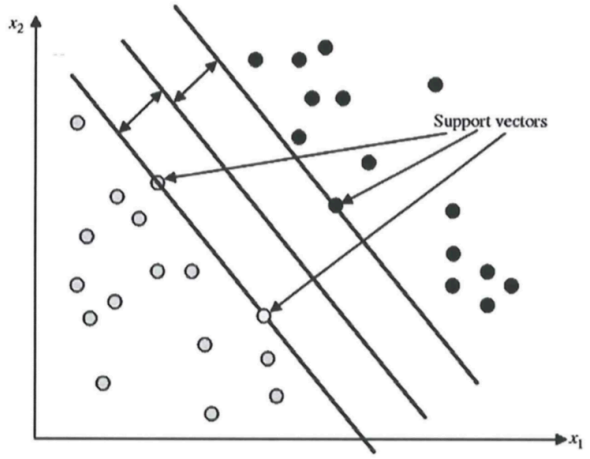

Figure 5: 训练集中的支持向量，中间的实线是决策边界。Picture source: Cynthia Rudin

因此我们先从解出的 $\boldsymbol{\alpha}^\star$ 中找到 $\alpha_i^\star > 0$ 对应的支持向量, 再从任一支持向量 $(\boldsymbol{x}_i, y_i)$ 处利用等式

$$y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) = 1 \tag{29}$$

计算出 $b^\star$.

由于训练集中的支持向量通常比较少，所以 $\boldsymbol{\omega}^\star = \sum_{i=1}^n \alpha_i^\star y_i \boldsymbol{x}_i$ 的计算非常快。

4 Nonseparable Case

在实际问题中经常碰到的情况是：不存在线性决策边界或超平面可以将训练集中的正负点区分开，如图6所示。因此需要对 SVM 模型(21)做一些修改以适用不可区分的情况。修改后的模型将允许一些分类错误，但需要为错误付出一定代价。

Figure 6: Nonseparable case.

修改后的 SVM 求解的优化问题变为：

$$\begin{aligned} &\min_{\boldsymbol{\omega},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{\omega}\|^2 + C\sum_{i=1}^{n}\xi_i \\ &\text{s.t. } y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i \\ &\xi_i \geq 0 \ i = 1, \ldots, n. \end{aligned} \tag{30}$$

可以看到(30)在限制条件中加入了一些 “松弛” (slack) ξ_i: 如果观察点 i 满足 $y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) \geq 1$, 令 $\xi_i = 0$ 可以避免惩罚；如果观察点 i 出现 $y_i(\boldsymbol{\omega}^\top \boldsymbol{x}_i + b) = 1 - \xi_i$ 且 $\xi_i > 0$，则需要付出代价 $C\xi_i$.

参数 C 代表对实现以下两个目标的权衡 (trade-off): (i) 保证训练集中大部分观察点的 margins 至少为 $\gamma = 1/\|\boldsymbol{\omega}\|$ (ii) 使 $\|\boldsymbol{\omega}\|$ 尽可能小。

在训练集可区分 (separable) 的情况下（正负点可以被超平面区分）, 在(30)中使用较大的 C 得到的决策边界与无松弛的优化问题(21)得到的决策边界很接近，如图7中实线所示，此时所有点的 margins 都是正的，但支持向量的 margins 很小。使用较小的 C 可以减小决策边界对 outliers 的敏感性：通过付出一些分类错误的代价保证大多数点的 margins 较大，如图7中虚线所示。

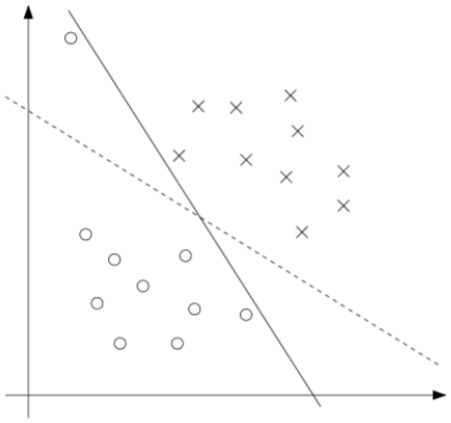

Figure 7: 不同的 C 对应的最优决策边界，其中实线对应较大的 C, 虚线对应较小的 C. Picture source: Andrew Ng

下面通过 KKT 条件求解(30), 建立 Lagrangian：

$$\mathcal{L}([\boldsymbol{\omega}, b, \boldsymbol{\xi}], \boldsymbol{\alpha}, \boldsymbol{r}) = \frac{1}{2}\boldsymbol{\omega}^\top\boldsymbol{\omega} + C\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i\left[-y_i(\boldsymbol{\omega}^\top\boldsymbol{x}_i + b) + 1 - \xi_i\right] + \sum_{i=1}^{n} r_i(-\xi_i) \tag{31}$$

其中 α_i's 和 r_i's 是 Lagrange multipliers. 令 $\mathcal{L}$ 关于 ξ_i 的一阶偏导数等于 0 得

$$\frac{\partial\mathcal{L}}{\partial\xi_i} = C - \alpha_i - r_i = 0 \implies \alpha_i^\star = C - r_i^\star. \tag{32}$$

由于 $r_i^\star \geq 0$, $\alpha_i^\star \geq 0$, 所以 $0 \leq \alpha_i^\star \leq C$, $i = 1, \ldots, n$.

此时 $\mathcal{L}$ 关于 $\boldsymbol{\omega}$ 和 b 的梯度与(23)-(24)相同，令其梯度等于零再代入 Lagrangian (31), 经过整理可得 $\boldsymbol{\alpha}^\star$ 是以下 dual problem 的解：

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \ & \sum_{i=1}^n \alpha_i - \frac{1}{2}\sum_{i=1}^n\sum_{k=1}^n \alpha_i\alpha_k y_i y_k \boldsymbol{x}_i^\top \boldsymbol{x}_k \\ \text{s.t. } & 0 \le \alpha_i \le C,\ i = 1, \dots, n \\ & \sum_{i=1}^n \alpha_i y_i = 0. \end{aligned} \tag{33}$$

(33)与(28)的唯一区别就是 α_i 的范围从 $\alpha_i \ge 0$ 变为 $0 \le \alpha_i \le C$.

此时截距项 $b^\star$ 的计算与之前的方法(29)不同。由 KKT 的 complementary slackness 条件可得

$$\begin{aligned} & y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) > 1 \ \Rightarrow\ \alpha_i^\star = 0 \\ & y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) < 1 \ \Rightarrow\ \xi_i^\star > 0 \ \Rightarrow\ r_i^\star = 0 \ \Rightarrow\ \alpha_i^\star = C \quad (\text{根据 (32)}) \\ & 0 < r_i^\star < C \ \Rightarrow\ 0 < \alpha_i^\star < C,\ \xi_i = 0 \ \Rightarrow\ y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) = 1 \end{aligned}$$

所以计算 $b^\star$ 只需找到 $0 < \alpha_i^\star < C$ 对应的观察点 $(\boldsymbol{x}_i, y_i)$, 然后利用等式 $y_i(\boldsymbol{\omega}^{\star\top}\boldsymbol{x}_i + b^\star) = 1$ 解出 $b^\star$.

下面介绍求解(33)的一个非常高效的算法 — sequential minimal optimization (SMO) 算法。

4.1 SMO 算法

SMO 算法本质上是一种坐标下降算法。假设有一组 α_i's 满足(33)中所有限制条件，如果固定 $\alpha_2, \dots, \alpha_n$, 通过调整 α_1 能使(33)的目标函数值上升吗? 答案是不能，由(33)的限制条件 $\sum_{i=1}^n \alpha_i y_i = 0$ 可得

$$\begin{aligned} \alpha_1 y_1 &= -\sum_{i=2}^n \alpha_i y_i \\ \alpha_1 &= -y_1 \sum_{i=2}^n \alpha_i y_i \quad (y_1 \in \{-1, 1\} \Rightarrow y_1 = 1/y_1) \end{aligned}$$

即当 $\alpha_2, \dots, \alpha_n$ 固定时，α_1 也被固定了。因此考虑同时更新 $\boldsymbol{\alpha}$ 中的 2 个元素，Platt (1998) 给出了一种启发式算法 (heuristics) 从 $\boldsymbol{\alpha}$ 中挑选要更新的一对 α_i 和 α_j. 假设固定 $\alpha_3, \dots, \alpha_n$，那么如何调整 α_1, α_2 使(33)的目标函数值上升?

首先由限制条件 $\sum_{i=1}^n \alpha_i y_i = 0$ 可得

$$\alpha_1 y_1 + \alpha_2 y_2 = -\sum_{i=3}^n \alpha_i y_i \triangleq \zeta. \tag{34}$$

又因为 $\alpha_1, \alpha_2 \in [0, C]$, 所以 (α_1, α_2) 只能位于正方形 $[0, C] \times [0, C]$ 内的一条线段上，如图8所示。从图8可以看到 α_2 的取值范围进一步缩小为 $[L, H]$. 在(34)中用 α_2 表示 α_1 得

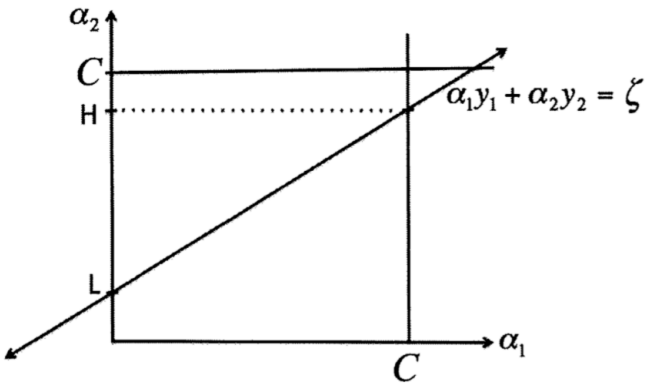

Figure 8: SMO 算法中固定 $\alpha_3, \ldots, \alpha_n$ 时 (α_1, α_2) 的取值范围。Picture source: Cynthia Rudin

$$\alpha_1 = y_1(\zeta - \alpha_2 y_2). \tag{35}$$

将(35)代入(33)的目标函数，此时的目标函数是 α_2 的二次函数 ($\alpha_3, \ldots, \alpha_n$ 固定)，很容易找到 α_2 在区间 $[L, H]$ 上的最优解，图9展示了一种情况。

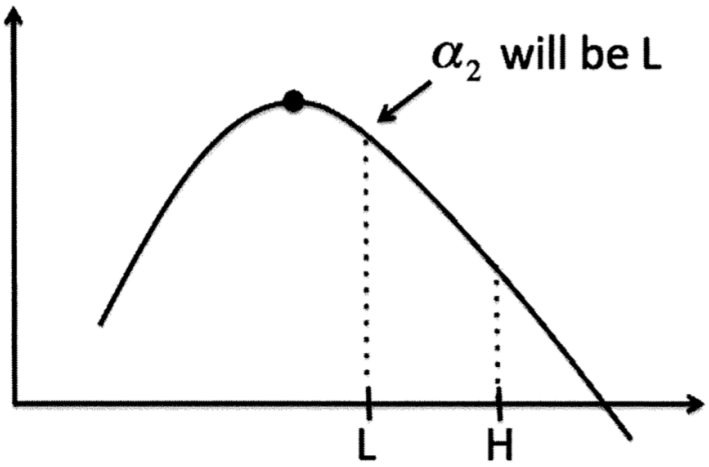

Figure 9: 固定 $\alpha_3, \ldots, \alpha_n$, 对(33)中的目标函数关于 α_2 做优化。Picture source: Cynthia Rudin

5 Kernels

最后介绍一种应用非常广泛的方法 — **kernel trick**, 它可以使 SVM 产生非常灵活的非线性决策边界或超曲面，如图10所示。SVM 与 kernels 的结合成为目前最强大的机器学习算法之一。

当我们在 $\boldsymbol{x}$ 的特征空间 (feature space) 上无法用线性决策边界将正负数据点区分时，一个解决办法是：将 $\boldsymbol{x}$ 所在的特征空间升维到一个 $\boldsymbol{\phi}(\boldsymbol{x})$ 所在的高维特征空间，使得在这个高维空间可以

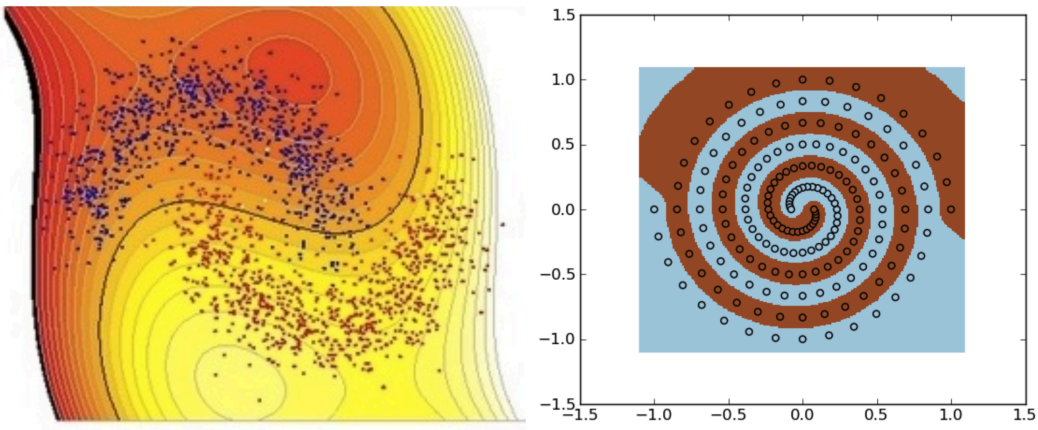

Figure 10: SVM 与 kernels 结合可以产生非常灵活的决策边界。Picture source: Cynthia Rudin

用线性超平面将正负点区分开，该超平面在原特征空间的投影就是一条可区分正负点的曲线边界，如图11所示。

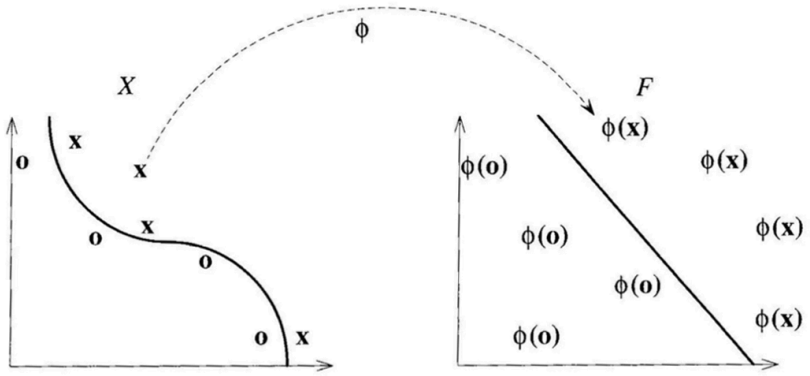

Figure 11: SVM 产生非线性决策边界的原理。Picture source: Cynthia Rudin

假设数据的特征空间是一维的，即 $x \in \mathbb{R}$, 我们想新增一些特征，比如加入 x^2, x^3, 此时用

$$\boldsymbol{\phi}(x) = \begin{pmatrix} x \\ x^2 \\ x^3 \end{pmatrix}$$

代表新的多维特征。然后用线性 SVM 分类器在 $\boldsymbol{\phi}(x)$ 所在的特征空间估计一个决策超平面。在

SVM 的优化问题中，我们需要计算如下的 dual problem:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \ & \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{k=1}^{n} \alpha_i \alpha_k y_i y_k \boldsymbol{x}_i^\top \boldsymbol{x}_k \\ \text{s.t. } & 0 \le \alpha_i \le C, \ i = 1, \ldots, n \\ & \sum_{i=1}^{n} \alpha_i y_i = 0. \end{aligned} \tag{36}$$

注意到(36)只用到了特征的内积 $\boldsymbol{x}_i^\top \boldsymbol{x}_k$，因此只需将(36)中的 $\boldsymbol{x}_i^\top \boldsymbol{x}_k$ 替换为 $\boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_k)$, 就得到了 $\boldsymbol{\phi}(\boldsymbol{x})$ 空间的 dual problem. 对每一个映射 $\boldsymbol{\phi}$, 定义它对应的 **kernel** 为:

$$K_\phi(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{\phi}(\boldsymbol{x})^\top \boldsymbol{\phi}(\boldsymbol{z}).$$

很多时候计算 kernel 的成本很小，但计算 $\boldsymbol{\phi}(\boldsymbol{x})$ 的成本却很高。比如在下面这个例子中，$\boldsymbol{x} \in \mathbb{R}^d$, 令

$$\boldsymbol{\phi}(\boldsymbol{x}) = \left(x_1^2, x_1 x_2, \ldots, x_1 x_d, x_2 x_1, \ldots, x_2 x_d, \ldots, x_d x_1, \ldots, x_d^2\right)^\top.$$

它的 kernel 为

$$\begin{aligned} K_\phi(\boldsymbol{x}, \boldsymbol{z}) &= \sum_{i=1}^{d} \sum_{j=1}^{d} (x_i x_j)(z_i z_j) \\ &= \left(\sum_{i=1}^{d} x_i z_i \right) \left(\sum_{j=1}^{d} x_j z_j \right) \\ &= \left(\boldsymbol{x}^\top \boldsymbol{z} \right)^2. \end{aligned}$$

可以看到计算 $\boldsymbol{\phi}(\boldsymbol{x})$ 的计算量为 $O(d^2)$, 而 $K_\phi(\boldsymbol{x}, \boldsymbol{z})$ 的计算量只有 $O(d)$.

如果 kernel 的形式为 $K_\phi(\boldsymbol{x}, \boldsymbol{z}) = \left(\boldsymbol{x}^\top \boldsymbol{z}\right)^r$，称其为 polynomial kernel, 它对应的 $\boldsymbol{\phi}(\boldsymbol{x})$ 中的每个元素都是一个 r 次多项式 $x_{i_1} x_{i_2} \ldots x_{i_r}$, $i_k \in \{1, \ldots, d\}$ ($\boldsymbol{x} \in \mathbb{R}^d$, $r < d$). 此时 $\boldsymbol{\phi}(\boldsymbol{x})$ 的计算量为 $O(d^r)$, 而 $K_\phi(\boldsymbol{x}, \boldsymbol{z})$ 的计算量仍为 $O(d)$. 因此从计算的角度，如果我们只需要 $K_\phi(\boldsymbol{x}, \boldsymbol{z})$ 的值，不一定要先计算出 $\boldsymbol{\phi}(\boldsymbol{x})$ 和 $\boldsymbol{\phi}(\boldsymbol{z})$.

- 对上述 polynomial kernel $K_\phi(\boldsymbol{x}, \boldsymbol{z}) = \left(\boldsymbol{x}^\top \boldsymbol{z}\right)^r$，如果不计算 $\boldsymbol{\phi}(\cdot)$ 在任意一点的值，对于一个测试点 $\boldsymbol{z}$, 如何用 $\boldsymbol{\phi}(\boldsymbol{x}_i)$ 所在空间的决策超平面预测其正负?

  假设 SVM 在 $\boldsymbol{x}$ 的特征空间的最优线性决策边界为

  $$\boldsymbol{\omega}^{\star\top} \boldsymbol{x} + b^\star = 0.$$

如果测试点 $\boldsymbol{z}$ 满足 $\boldsymbol{\omega}^{\star\top}\boldsymbol{z}+b^\star \geq 0$, 预测 $\boldsymbol{z}$ 点的 y 值为正, 反之为负。由 Lagrangian stationarity (23)可得 $\boldsymbol{\omega}^\star = \sum_{i=1}^n \alpha_i^\star y_i \boldsymbol{x}_i$, 则最优决策边界可写为:

$$\sum_{i=1}^n \alpha_i^\star y_i \boldsymbol{x}_i^\top \boldsymbol{x} + b^\star = 0. \tag{37}$$

注意到(37)只用到点的内积 $\boldsymbol{x}_i^\top \boldsymbol{x}$, $i = 1, \ldots, n$. 因此在 $\boldsymbol{\phi}(\boldsymbol{x})$ 的空间中, 最优决策超平面应具有以下形式:

$$\sum_{i=1}^n \alpha_i^\star y_i \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}) + b^\star = 0.$$

使用 kernel 表示:

$$\sum_{i=1}^n \alpha_i^\star y_i K_\phi(\boldsymbol{x}_i, \boldsymbol{x}) + b^\star = 0 \tag{38}$$

其中 $b^\star$ 可以从某个 $0 < \alpha_j^\star < C$ 对应的观察点 $(\boldsymbol{\phi}(\boldsymbol{x}_j), y_j)$ 处计算得到:

$$b^\star = y_j - \boldsymbol{\omega}^{\star\top}\boldsymbol{\phi}(\boldsymbol{x}_j) = y_j - \sum_{i=1}^n \alpha_i^\star y_i \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j) = y_j - \sum_{i=1}^n \alpha_i^\star y_i K_\phi(\boldsymbol{x}_i, \boldsymbol{x}_j).$$

方程(38)在 $\boldsymbol{x}$ 所在的空间一般对应一条曲线或曲面。如果在(38)中使用 polynomial kernel, 在 $\boldsymbol{x}$ 的空间就得到了一条多项式决策边界。

对于测试点 $\boldsymbol{z}$, 如果 $\sum_{i=1}^n \alpha_i^\star y_i K_\phi(\boldsymbol{x}_i, \boldsymbol{z}) + b^\star \geq 0$, 预测 $\boldsymbol{z}$ 的 y 值为正, 反之为负。

以上分析表明: 只需要定义一个 kernel function $K(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ 就可以得到 SVM 的决策边界, 甚至不需要知道映射 $\boldsymbol{\phi}(\cdot)$ 的具体形式。

SVM 中一个常用的 kernel function 是 **Gaussian kernel**:

$$K(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{z}\|^2}{2\sigma^2}\right). \tag{39}$$

Gaussian kernel 的值 $K(\boldsymbol{x}, \boldsymbol{z})$ 反映点 $\boldsymbol{x}$ 和 $\boldsymbol{z}$ 的相似度: 当点 $\boldsymbol{x}$ 和 $\boldsymbol{z}$ 很接近时, $K(\boldsymbol{x}, \boldsymbol{z})$ 的值较大; 当点 $\boldsymbol{x}$ 和 $\boldsymbol{z}$ 相距较远时, $K(\boldsymbol{x}, \boldsymbol{z})$ 的值较小。那么如何证明确实存在一个映射 $\boldsymbol{\phi}(\cdot)$ 使得 $K(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{\phi}(\boldsymbol{x})^\top \boldsymbol{\phi}(\boldsymbol{z})$?

先来考察一个有效的 kernel 需要具备的必要条件。如果存在映射 $\boldsymbol{\phi}(\cdot)$ 使得 kernel function $K(\boldsymbol{x}, \boldsymbol{z}) = \boldsymbol{\phi}(\boldsymbol{x})^\top \boldsymbol{\phi}(\boldsymbol{z})$, 则

$$K(\boldsymbol{z}, \boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{z})^\top \boldsymbol{\phi}(\boldsymbol{x}) = \boldsymbol{\phi}(\boldsymbol{x})^\top \boldsymbol{\phi}(\boldsymbol{z}) = K(\boldsymbol{x}, \boldsymbol{z}).$$

即 $K(\cdot,\cdot)$ 需要满足对称关系 $K(\boldsymbol{x}, \boldsymbol{z}) = K(\boldsymbol{z}, \boldsymbol{x})$.

其次，对 $\mathbb{R}^d$ 上的任意 n 个点 $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, 定义矩阵

$$\mathcal{K} = (K_{ij})_{n\times n} \tag{40}$$

其中 $K_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$. 如果 $K(\cdot,\cdot)$ 是一个有效的 kernel, 则矩阵 $\mathcal{K}$ 是一个对称矩阵且存在映射 $\boldsymbol{\phi}(\cdot)$ 使得 $\mathcal{K}$ 的每个元素 $K_{ij} = \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j)$. 此时对于任意 $\boldsymbol{z} \in \mathbb{R}^d$,

$$\begin{aligned}
\boldsymbol{z}^\top \mathcal{K} \boldsymbol{z} &= \sum_{i=1}^d \sum_{j=1}^d z_i K_{ij} z_j \\
&= \sum_{i=1}^d \sum_{j=1}^d z_i z_j \boldsymbol{\phi}(\boldsymbol{x}_i)^\top \boldsymbol{\phi}(\boldsymbol{x}_j) \\
&= \left(\sum_{i=1}^d z_i \boldsymbol{\phi}(\boldsymbol{x}_i)\right)^\top \left(\sum_{j=1}^d z_j \boldsymbol{\phi}(\boldsymbol{x}_j)\right) \\
&= \left(\sum_{i=1}^d z_i \boldsymbol{\phi}(\boldsymbol{x}_i)\right)^\top \left(\sum_{i=1}^d z_i \boldsymbol{\phi}(\boldsymbol{x}_i)\right) \\
&\geq 0
\end{aligned}$$

因此矩阵 $\mathcal{K}$ 是一个半正定矩阵。下面的定理表明上述条件不仅是必要条件也是充分条件。

Theorem 3 (Mercer)**.** *函数 $K(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ 是一个有效 kernel 的充分必要条件是：对 $\mathbb{R}^d$ 上的任意有限个点 $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, 由(40)定义的矩阵 $\mathcal{K}$ 是一个对称半正定矩阵。*

Mercer 定理保证了 Gaussian kernel (39)是一个有效的 kernel, 事实上 Gaussian kernel 对应的映射 $\boldsymbol{\phi}(\cdot)$ 将原特征映射到一个无穷维空间。

Kernel 的应用不仅限于 SVM. 只要一个算法仅用到特征的内积 $\boldsymbol{x}^\top \boldsymbol{z}$，就可以将其替换为一个 kernel $K(\boldsymbol{x}, \boldsymbol{z})$，从而能在更高维的空间继续使用该算法，这个方法被称为 kernel trick.

References

Platt, J. (1998). Sequential minimal optimization: A fast algorithm for training support vector machines.

