北航《数据挖掘导论》课程讲义

王静远

北京航空航天大学计算机学院

2019 年 10 月 26 日

1 Support Vector Machines

支持向量机 (Support Vertor Machines, SVM) 是一种常用的二分类模型，其基本思想是构建特征空间上类间间隔最大的超平面，并以此进行分类 [10]。本章首先介绍 SVM 的分类间隔定义，并给出基础 SVM 的目标函数，以及其对应的凸二次规划问题。通过使用拉格朗日乘子法，可导出其对偶问题，并给出 SVM 目标函数的求解方法。其次，针对基础 SVM 只能处理线性可分数据的情况，引入软间隔 (soft margin) 及核方法 (kernel method) 的概念。对于近似线性可分数据，通过引入软间隔来允许 SVM 在少数样本下不满足约束条件，达到分类的目的。对于线性不可分数据，可使用核方法，将原始数据空间映射到高维特征空间中，并在高维特征空间下构造最优超平面，实现数据的分类。最后，将介绍一种快速的启发式学习方法——序列最小优化算法 (SMO)，该方法是目前较为常用的 SVM 训练算法。

1.1 SVM 的分类间隔

SVM 的目的是在数据的特征空间中构建一个超平面，该超平面可以将特征空间划分为两个空间，位于平面 “上” 半部分的数据为正类，位于平面 “下” 半部分的数据为负类。设超平面方程的表达式为：

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

则给定任意一个样本 $\{\boldsymbol{x}_i, y_i\}$，SVM 的判别函数表示为

$$f(\boldsymbol{x}_i) = \text{sign}\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) = \begin{cases} 1, & \text{if } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 0 \\ -1, & \text{if } \boldsymbol{w}^\top \boldsymbol{x}_i + b < 0 \end{cases} \tag{2}$$

其中 $f(\boldsymbol{x}_i) = 1$ 表示分为正类，$f(\boldsymbol{x}_i) = -1$ 表示分为负类。样本点 $\boldsymbol{x}_i$ 到分类面的距离用来度量 SVM 对于该次分类的置信度，即

$$g(\boldsymbol{x}_i) = \left|\boldsymbol{w}^\top \boldsymbol{x}_i + b\right|. \tag{3}$$

对于一个样本 $(\boldsymbol{x}_i, y_i)$，我们可以认为 SVM 超平面对于该样本分类的正确程度 $\hat{\gamma}_i$ 为

$$\begin{aligned} \hat{\gamma}_i &= y_i \cdot f(\boldsymbol{x}_i) \cdot g(\boldsymbol{x}_i) \\ &= y_i \cdot \text{sign}\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \cdot \left|\boldsymbol{w}^\top \boldsymbol{x}_i + b\right| \\ &= y_i \cdot (\boldsymbol{w}^\top \boldsymbol{x}_i + b). \end{aligned} \tag{4}$$

$\hat{\gamma}_i$ 又被称为**函数间隔 (functional margin)**.

使用**函数间隔**作为样本分类效果的度量存在一个问题就是尺度的不确定。举例来说，如果我们将 $\boldsymbol{w}$ 和 b 改为 $2\boldsymbol{w}$ 和 $2b$，超平面的表达式由 $\boldsymbol{w}\boldsymbol{x}+b=0$ 变为了 $2\boldsymbol{w}\boldsymbol{x}+2b=0$。超平面本身没有发生变化，但是 $\hat{\gamma}_i$ 变为了之前的 2 倍，这显然不够合理。为此，我们将超平面的表达式归一化为

$$\frac{\boldsymbol{w}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{w}\|} = 0. \tag{5}$$

$\hat{\gamma}_i$ 可被重写为

$$\gamma_i = y_i \cdot \left(\frac{\boldsymbol{w}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{w}\|} \right). \tag{6}$$

γ_i 被称为**几何间隔 (geometric margin)**。

1.2 SVM 的分类目标函数

SVM 的优化目标是最大化 γ_i 的最小值。通俗的讲就是让数据集中 “离分类面最近的点” 离分类面尽可能远。具体来讲，给定数据集 $T=\{(\boldsymbol{x}_i, y_i)\}, i \in 1\ldots, N$，数据集中样本到分类面的最小值为

$$\gamma = \min_{i=1,\ldots,N} \gamma_i. \tag{7}$$

SVM 的优化目标是最大化 γ，即

$$\arg\max_{\boldsymbol{w},b} \quad \gamma(\boldsymbol{w}, b). \tag{8}$$

对这样一个优化目标，我们可以进一步写成如下的**带约束的最优化问题**：

$$\arg\max_{\boldsymbol{w},b} \quad \gamma \tag{9}$$

$$\text{s.t.} \quad \gamma_i \geq \gamma, i = 1, 2, \ldots, N. \tag{10}$$

在公式 Eq. (48)中，我们要求对所有的 i，$\gamma_i \geq \gamma$。也就是说 γ 是 γ_i 中的最小值。把 Eq. (7)带入到 Eq.(48)中，可得

$$\arg\max_{\boldsymbol{w},b} \quad \gamma \tag{11}$$

$$\text{s.t.} \quad y_i \left(\frac{\boldsymbol{w}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{w}\|} \right) \geq \gamma, i = 1, 2, \ldots, N. \tag{12}$$

为了方便讨论，我们将公式中的 γ 转化为函数间隔，可得如下带约束的目标函数

$$\arg\max_{\boldsymbol{w},b} \quad \frac{\hat{\gamma}}{\|\boldsymbol{w}\|} \tag{13}$$

$$\text{s.t.} \quad y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b \right) \geq \hat{\gamma}, i = 1, 2, \ldots, N. \tag{14}$$

进一步地，我们可以看出 $\hat{\gamma}$ 的大小对于求解整个问题没有影响，我们可以令

$$\boldsymbol{w}' = \frac{\boldsymbol{w}}{\hat{\gamma}}, \quad b' = \frac{b}{\hat{\gamma}}. \tag{15}$$

则可以获得

$$\arg\max_{\boldsymbol{w},b} \quad \frac{1}{\|\boldsymbol{w}'\|} \tag{16}$$

$$\text{s.t.} \quad y_i \left(\boldsymbol{w}'^\top \boldsymbol{x}_i + b' \right) \geq 1, i = 1, 2, \ldots, N. \tag{17}$$

同时，最大化 $\frac{1}{\|\boldsymbol{w}\|}$ 等价于最小化 $\frac{1}{2}\|\boldsymbol{w}\|^2$。SVM 的目标函数转化为如下带约束的二次优化问题

$$\arg\min_{\boldsymbol{w},b} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 \tag{18}$$

$$\text{s.t.} \quad y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) - 1 \geq 0, i = 1, 2, \ldots, N. \tag{19}$$

这是一个凸二次规划问题 (convex quadratic programming problem)。对于这类带约束的二次规划问题，典型的求解方法是使用拉格朗日乘子法通过求解其对偶问题进行解决。下小节中我们将介绍拉格朗日乘子法的的相关内容。

1.3 拉格朗日乘子法

1.3.1 原始问题

公式 Eq. (47)的目标函数可以用拉格朗日乘子法进行求解。在这里我们简单介绍一下拉格朗日乘子法。对于带约束的优化问题，如

$$\min_w \quad f(w) \tag{20}$$

$$\text{s.t.} \quad g_i(w) \leq 0,\ i = 1, \ldots, N, \tag{21}$$

$$h_j(w) = 0,\ j = 1, \ldots, M. \tag{22}$$

我们引入拉格朗日乘子 (Lagrange multiplier)，α_i，β_j，将原问题表达为求解目标与约束的线性组合，即

$$L(w, \alpha, \beta) = f(w) + \sum_i \alpha_i g_i(w) + \sum_j \beta_j h_j(w), \alpha_i \geq 0. \tag{23}$$

定义函数

$$\theta_P(w) = \max_{\alpha_i \geq 0, \beta_j} L(w, \alpha, \beta). \tag{24}$$

容易证明，

$$\theta_P(w) = \begin{cases} f(w), & \text{if } g_i(w) \leq 0,\ h_j(w) = 0 \\ +\infty, & \text{other} \end{cases} \tag{25}$$

当 w 满足原问题约束时，原优化问题 E.q. (20)可以写为

$$\min_w \theta_P(w) = \min_w \max_{\alpha_i \geq 0, \beta_j} L(w, \alpha, \beta). \tag{26}$$

请注意，这里 $\theta_P(w) = \max_{\alpha_i \geq 0, \beta_j} L(w, \alpha, \beta)$ 是一个 w 的函数。Eq. (26)通过选择合适的 w 使得 $\theta_P(w)$ 最小化。

所以 Eq. (26)，等价于 Eq. (20)中定义的优化问题。Eq. (26)又被称之为**广义拉格朗日函数的极小极大问题**。这里极小极大的意思是最小化 L 函数的极大值。

1.3.2 对偶问题

对于 Eq. (26)中给出的原始问题，我们定义一个对偶问题：

$$\max_{\alpha_i \geq 0, \beta_j} \theta_D(\alpha, \beta) = \max_{\alpha_i \geq 0, \beta_j} \min_w L(w, \alpha, \beta). \tag{27}$$

这里 $\theta_D(\alpha,\beta)=\min_w L(w,\alpha,\beta)$ 是 α,β 的函数。是通过调整 α,β 使得 $\theta_D(\alpha,\beta)$ 取得极大值。Eq. (27)的结构和原始问题的定义正好是相反的，所以叫对偶问题。Eq. (27)被成为是**广义拉格朗日函数的极大极小问题**。这里极大极小的意思是最大化 L 函数的极小值。

直观上，原始问题和对偶问题有一定的联系，一个是最大化最小值，一个是最小化最大值，但很显然这两个问题并不等价。这就好比一个是在房价最贵的城市买最便宜的房子，一个是在房价最便宜的城市里买最贵的别墅，二者并不等价。在广义拉格朗日函数中，二者满足如下定理关系。

定理 1: 若原始问题和对偶问题都有最优值，则

$$d^\star=\max_{\alpha_i\geq 0,\beta_j}\min_w L(w,\alpha,\beta)\leq\min_w\max_{\alpha_i\geq 0,\beta_j}L(w,\alpha,\beta)=p^\star. \tag{28}$$

证明: 对于任意的 α,β,w，有

$$\theta_D(\alpha,\beta)=\min_w L(w,\alpha,\beta)\leq L(w,\alpha,\beta)\leq\max_{\alpha_i\geq 0,\beta_j}L(w,\alpha,\beta)=\theta_P(w). \tag{29}$$

即对于任意的 α,β,w，有

$$\theta_D(\alpha,\beta)\leq\theta_P(w). \tag{30}$$

由于原始问题和对偶问题均有最优解，所以，

$$\max_{\alpha_i\geq 0,\beta_j}\theta_D(\alpha,\beta)\leq\min_w\theta_P(w). \tag{31}$$

即

$$d^\star=\max_{\alpha_i\geq 0,\beta_j}\min_w L(w,\alpha,\beta)\leq\min_w\max_{\alpha_i\geq 0,\beta_j}L(w,\alpha,\beta)=p^\star. \tag{32}$$

■

在定理 1 中，$p^\star$ 和 $d^\star$ 分别是原问题和对偶问题的最优值。根据定理 1 可知，对偶问题的最优值是原问题最优值的一个下界。这个性质叫做弱对偶性 (weak duality)，对于所有的优化问题都成立。

1.3.3 KKT 条件

既然有弱对偶性，那必然有强对偶性 (strong duality)，强对偶性指的原始问题和对偶问题的最优解严格相等，即:

$$d^\star=L(w^\star,\alpha^\star,\beta^\star)=p^\star, \tag{33}$$

其中 $w^\star$ 和 $\alpha^\star,\beta^\star$ 分别是原问题和对偶问题的最优解。在强对偶性成立的情况下，我们就可以通过对原始问题的对偶问题的求解来得到最优解（SVM 就是这么做的），但并不是所有情况下强对偶性都成立，它会有一定的前提。这个前提条件就是 KKT 条件。

定理 2 假设 $f(w),g(w)$ 是凸函数，$h(w)$ 是仿射函数，可行域中至少有一点使得不等式约束 $g(w)\leq 0$ 严格成立。那么 $w^\star$ 和 $\alpha^\star,\beta^\star$ 分别是原问题和对偶问题的最优解的充分必要条件是 $w^\star,\alpha^\star,\beta^\star$ 满足下述 Karush-Kuhn-Tucker(KKT) 条件:

条件 *1*，*Stationarity:*

$$\nabla_w f(w^\star)+\sum_i\alpha_i^\star\nabla_w g_i(w^\star)+\sum_j\beta_j^\star\nabla_w h_j(w^\star)=0. \tag{34}$$

条件 *2*，*Primal feasibility:*

$$g_i(w^\star)\leq 0,\ i=1,\ldots,N, \tag{35}$$

$$h_j(w^\star)=0,\ j=1,\ldots,M. \tag{36}$$

条件 *3*，*Dual feasibility:*

$$\alpha_i^\star \geq 0, \text{ for } i = 1, \ldots, N. \tag{37}$$

条件 *4*，*Complementary slackness* **(互补松弛)**:

$$\alpha_i^\star g_i(w^\star) = 0, \text{ for } i = 1, \ldots, N. \tag{38}$$

这里我们先对这 4 个条件进行直观性的解释，来帮助大家理解 KKT 条件背后的几何含义。

首先，条件 2 和条件 3 分别是原问题和对偶问题的可行域，如果不能够满足那么 $w^\star$ 和 $\alpha^\star$ 就不是原问题和对偶问的解了，所以肯定是要满足的。因此这两个条件被称为可行 (feasibility) 条件。

条件 1 则是通过求解如下式子获得的：

$$\left.\frac{\partial L(w, \alpha, \beta)}{\partial w}\right|_{w=w^\star} = 0 \tag{39}$$

由于 $L(w, \alpha, \beta)$ 是一个凸函数，所以无论在原问题还是对偶问题中，Eq. (39)都是应该满足的。

如何理解由 Eq. (39)得出的条件 1 呢？这里可以分为两种情况讨论。第一种情况是等式约束的情况，即

$$\arg\min_w \quad f(w) \tag{40}$$

$$\text{s.t.} \quad h_j(w) = 0, \ j = 1, \ldots, M. \tag{41}$$

此时，Eq. (34)可以表达为

$$\nabla_w f(w^\star) = -\sum_j \beta_j \nabla_w h_j(w^\star). \tag{42}$$

这个表达式该如何理解呢？其直观的含义是当 w 取得最优解的时候，约束函数 $h(w^\star) = 0$ 和目标函数 $y = f(w^\star)$ 的某一条等高线是相切的，或者说在两者在 $w^\star$ 点的**切线是平行的**。

可以想象这样一个场景，当你沿着一条道路从山顶走到山谷。这里山的等高线就是你要优化的目标函数 $f(w)$，你要下山下到最低处，也就是使 $f(w)$ 尽可能的小。由于你必须沿着一条固定的道路下山，即约束 $h(w) = 0$，所以当你沿着道路下到山的最底处的时候，**必然是道路和山的等高线相切的地方**，（否则就一定能够沿着道路继续下山）。所以对于等式约束的情况，条件 1 也是需要满足的。

继续考虑不等式约束的情况，即

$$\arg\min_w \quad f(w) \tag{43}$$

$$\text{s.t.} \quad g_i(w) \leq 0, \ i = 1, \ldots, N. \tag{44}$$

此时，Eq. (34)可以表达为

$$\nabla_w f(w^\star) = -\sum_i \alpha_i^\star \nabla_w g_i(w^\star). \tag{45}$$

对于 $g_i(w) \leq 0$ 的不等式约束，当 $g_i(w^\star) = 0$ 时，其情况和等式约束相同，Eq. (34)解释为 $g_i(w^\star) = 0$ 与等高线相切。当 $g_i(w^\star) < 0$ 时（根据 KKT 条件的定义这种情况是必然存在的），$f(w^\star)$ 的最优解出现在 $\nabla_w f(w^\star) = 0$ 的位置。

还用下山举例子，不等式约束相当于我们在山坡上修了一圈篱笆，我们只能在篱笆内下到尽可能底的位置。如果篱笆圈到了半山腰上（山谷在篱笆外），最低点必然出现在篱笆墙的边界上，最低点上篱笆墙和山的等高线相切。如果篱笆修在山谷外，那么最低点就是山谷等高线的最低点，等于是一个无约束

的情况，此时最优解出现在 $\nabla_w f(w^\star) = 0$ 的位置。这时，最优解已经和 $g_i(w)$ 无关了，无论 $g_i(w)$ 和 $\nabla_w g_i(w)$ 取何值，$\nabla_w f(w^\star) = 0$。满足这种条件只有一种情况，也就是 $\alpha_i^\star = 0$。

通过上面的分析，我们可以看出，要想使 Eq. (45)成立，那么 $g_i(w^\star) = 0$ 和 $\alpha_i^\star = 0$ 两者必须满足其一。最优解或者在不等式约束的边界达到，或者不等式约束不发挥作用。这就引出了 KKT 条件的第 4 个条件，即 KKT 对偶的互补松弛：

$$\alpha_i^\star g_i(w^\star) = 0, \text{ for } i = 1, \ldots, N. \tag{46}$$

在稍后的 SVM 介绍中，我们会发现这个互补松弛条件直接解释了为什么 SVM 模型中只有支持向量 (support vector) 是发挥作用的。

1.4 SVM 目标函数的求解

重新回到 SVM 目标函数的求解问题，如下面的公式所示

$$\arg\min_{\boldsymbol{w},b} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 \tag{47}$$

$$\text{s.t.} \quad y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) - 1 \geq 0, i = 1, 2, \ldots, N. \tag{48}$$

引入拉格朗日乘子可得

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_i \alpha_i \left(y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) - 1\right). \tag{49}$$

原问题的无约束形式为

$$\min_{\boldsymbol{w},b} \max_{\alpha_i \geq 0} L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = p^\star. \tag{50}$$

进一步，我们将问题转化为对偶问题：

$$\max_{\alpha_i \geq 0} \min_{\boldsymbol{w},b} L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = d^\star. \tag{51}$$

容易验证，在该问题中 KKT 条件是成立的。因此，我们让 $L(\boldsymbol{w}, b, \boldsymbol{\alpha})$ 函数对于 $\boldsymbol{w}, b$ 最小化，可得

$$\frac{\partial L}{\partial w} = 0 \Rightarrow \boldsymbol{w} = \sum_i \alpha_i y_i \boldsymbol{x}_i, \tag{52}$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_i \alpha_i y_i = 0. \tag{53}$$

将这两个式子回带到 L 函数中，可得

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \sum_{i=1}^N a_i - \frac{1}{2}\sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{54}$$

进一步地，求解对偶问题外层优化，可得如下带约束的优化问题：

$$\arg\max_{\boldsymbol{\alpha}} \quad \sum_{i=1}^N a_i - \frac{1}{2}\sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j \tag{55}$$

$$\text{s.t.} \quad a_i \geq 0, i = 1, 2, \ldots, N \tag{56}$$

$$\sum_{i=1}^N \alpha_i y_i = 0. \tag{57}$$

这是一个二次规划 (quadratic programming)，标准的解法是使用序列最小优化算法 (Sequential Minimal Optimization, SMO)。

Eq. (55)中的优化问题同原优化问题等价，该公式是拉格朗日乘子 α_i 的方程。Eq. (52)中的红色部分，给出了 α_i 和 SVM 分类界面权重 $\boldsymbol{w}$。将其带入到 SVM 的分类界面公式当中，可以获得

$$s(x) = \left(\sum_{i=1}^{N} \alpha_i y_i \boldsymbol{x}_i\right)^{\top} \boldsymbol{x} + b \tag{58}$$

$$= \sum_{i=1}^{N} \alpha_i y_i \langle \boldsymbol{x}_i, \boldsymbol{x} \rangle + b. \tag{59}$$

在该式子中 $\langle \boldsymbol{a}, \boldsymbol{b} \rangle$ 表示两个向量的内积。在该式子中，我们可以看出，SVM 最优分类界面等价于将待预测样本 $\boldsymbol{x}$ 同所有训练样本求内积，然后用 $\alpha_i y_i$ 进行加权组合。两个向量的内积正比于他们之间的 cosin 距离，所以 SVM 是实际上是测算了待预测样本同每一个训练样本的距离，然后根据训练样本的标注做出了最终的判断。

在真实的分类过程中，不需要进行这么复杂的计算。一方面，我们可以提前算出来 $\boldsymbol{w}$。另一方面，根据 KKT 条件，我们可以得到 SVM 一个非常有趣的性质。SVM 的 KKT 条件为:

$$\begin{cases} \alpha_i \geq 0, \\ y_i s(\boldsymbol{x}_i) \geq 1, \\ \alpha_i (y_i s(\boldsymbol{x}_i) - 1) = 0. \end{cases} \tag{60}$$

其中的第三项就是互补松弛。在这一项中，α_i 和 $y_i s(\boldsymbol{x}_i) = 1$ 有如下关系

$$\begin{aligned} y_i s(\boldsymbol{x}_i) > 1 \Rightarrow \quad \alpha_i = 0, \\ y_i s(\boldsymbol{x}_i) = 1 \Rightarrow \quad \alpha_i > 0. \end{aligned} \tag{61}$$

也就是说，只有对处在分类界面边界上的训练样本，α_i 才不等于 0，否则 $\alpha_i = 0$，这个样本没有价值。这些处在分类界面边界上的训练样本就是所谓的**支持向量 support vector**。

1.5 软间隔

对于线性可分数据，使用 Eq. (47)作为目标函数优化的 SVM 即可完成对样本的正负例分类。但在实际数据中，完全线性可分的情况往往较少。一种常见的情况是近似线性可分的数据，即数据本身是线性结构的，但存在少部分样本无法满足优化目标中函数间隔大于等于 1 的约束条件，通常这种情况来自于数据中的噪声，这类样本被称为特异点 (outlier)。

为将 SVM 扩展到近似线性可分问题，可引入“软间隔”(soft margin) 的概念，允许支持向量机在一些样本上不满足约束 [2]。具体地说，对每个样本点 $(\boldsymbol{x}_i, y_i)$ 引入一个松弛变量 (slack variable) $\xi_i \geq 0$，表示该数据点允许偏离的函数间隔大小。故新的约束条件为

$$y_i(\boldsymbol{w}^{\top} \boldsymbol{x_i} + b) \geq 1 - \xi_i, \quad i = 1, 2, ..., N \tag{62}$$

同时，在目标函数中引入增加松弛变量带来的代价，新的目标函数为

$$\frac{1}{2}\|\boldsymbol{w}\|^2 + C \sum_{i=1}^{N} \xi_i \tag{63}$$

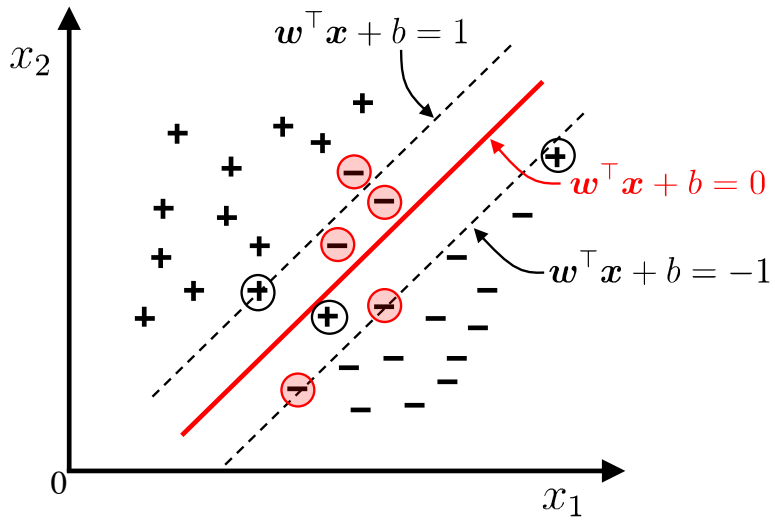


图 1: 软间隔示例

此处参数 C 为一个正实数，用于调节目标函数中控制“寻找间隔最大的分类面”和“降低分类误差”两项的相对权重大小。C 越大，对分类误差的容忍程度越低；当 $C \to \infty$ 时，目标函数退化为普通的 SVM。在实际应用中，C 值往往根据具体问题由用户自行定义。图1展示了使用软间隔的支持向量机分类结果。

增加了软间隔后，原始问题可写为

$$\min_{\boldsymbol{w},b,\xi} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i \tag{64}$$

$$\text{s.t.} \quad y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \tag{65}$$

$$\xi_i \geq 0, \quad i = 1, 2, \ldots, N. \tag{66}$$

使用之前的方法，构造该问题的拉格朗日函数，得到

$$L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\delta}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i(y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) - 1 + \xi_i) - \sum_{i=1}^{N}\delta_i\xi_i \tag{67}$$

其中，$\alpha_i \geq 0, \delta_i \geq 0$。原问题的无约束形式为

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \max_{\alpha_i \geq 0, \delta_i \geq 0} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\delta}) = p^\star. \tag{68}$$

进一步，我们将问题转化为对偶问题：

$$\max_{\alpha_i \geq 0, \delta_i \geq 0} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\delta}) = d^\star. \tag{69}$$

容易验证，在该问题中 KKT 条件是成立的。因此，我们让 $L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\delta})$ 函数对于 $\boldsymbol{w}, \boldsymbol{\xi}, b$ 最小化，可得

$$\frac{\partial L}{\partial w} = 0 \Rightarrow \boldsymbol{w} = \sum_i \alpha_i y_i \boldsymbol{x}_i, \tag{70}$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_i \alpha_i y_i = 0, \tag{71}$$

$$\frac{\partial L}{\partial \xi_i} = 0 \Rightarrow C - \alpha_i - \delta_i = 0 \Rightarrow 0 \leq \alpha_i \leq C. \tag{72}$$

将 $\boldsymbol{w}$ 回带到 $L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\delta})$ 函数中并化简，可得

$$L(\boldsymbol{w}, \boldsymbol{\xi}, b, \boldsymbol{\alpha}, \boldsymbol{\delta}) = \sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{73}$$

因此，带软间隔的线性支持向量机的对偶问题为

$$\max_{\boldsymbol{\alpha}} \quad \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j \tag{74}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} \alpha_i y_i = 0, \tag{75}$$

$$0 \leq \alpha_i \leq C, i = 1, 2, \ldots, N. \tag{76}$$

可以看到，增加了软间隔的 SVM 优化问题与原问题 Eq. (55)基本相同，其区别在于对 α_i 增加了一个新的约束条件，即 $\alpha_i \leq C$。因此，该问题同样可用 SMO 算法进行求解。

1.6 核方法

上文介绍的 SVM 算法对于解决线性可分或近似线性可分问题是一种十分有效的分类方法。但是，在现实中，并不是所有数据都是线性可分或近似线性可分的。这时，就需要利用核方法 (Kernel Method) 来构建解决非线性问题的支持向量机。核方法的基本思想是通过使用核函数 (Kernel Function)，将数据由原始特征空间映射到高维隐式特征空间中，并在高维空间上使用线性决策边界对数据进行划分。需要知道的是，核方法与非线性 SVM 并不是绑定关系，实际上，核方法的出现要早于 SVM，并应用在了多种统计学习问题中。本节将介绍如何使用核函数来构建非线性 SVM，也称核 SVM (Kernel SVM)。

1.6.1 非线性问题的高维转换

在讨论非线性问题前，需要首先了解线性可分问题的定义。一般来说，对于给定 n 维欧几里得空间上的两类点 C_1, C_2，若存在一组实数 $w_1, w_2, ..., w_n, k$，满足

$$\sum_{i=1}^{n} w_i x_i \begin{cases} > k, & \text{if } \boldsymbol{x} \in C_1 \\ < k, & \text{if } \boldsymbol{x} \in C_2 \end{cases}$$

则称 C_1, C_2 是线性可分的，而

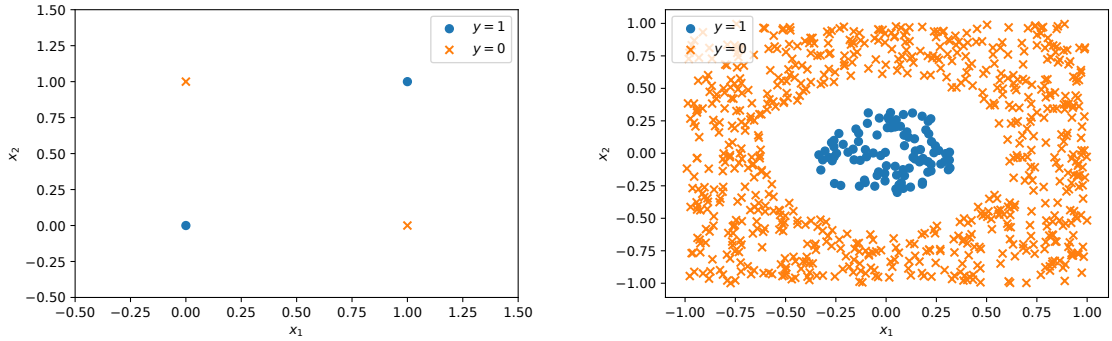
$$\sum_{i=1}^{n} w_i x_i = k$$

即为该组数据的分类超平面。反之，若无法找到一个超平面，满足以上分类条件，但能够从 n 维欧几里得空间中找到一个超曲面，将 C_1, C_2 两类数据分开，则称 C_1, C_2 为非线性可分数据。针对非线性可分数据的分类问题即为非线性问题。

对于非线性问题，一个经典而简单的例子是计算机中的异或函数 (XOR)。二元异或函数接受两个输入变量 $x_1, x_2 \in \{0, 1\}$, 该函数定义为

$$f(x_1, x_2) = \begin{cases} 1, & \text{if } x_1 \neq x_2 \\ 0, & \text{if } x_1 = x_2 \end{cases}$$

图2 (a) 展示了二元异或函数在二维平面上的分布形式，其中，$y = 0$ 的点使用符号"○" 表示，$y = 1$ 的点使用符号"×" 表示。显然，我们无法在图中找到一个超平面，将 $y = 0$ 与 $y = 1$ 两个类准确分开。但



(a) 异或函数 (b) 一个更复杂的例子

图 2: 线性不可分数据示例

可以使用一条椭圆曲线将两类划分开，因此，这是一个非线性可分问题。此外，图2 (b) 展示了一个更为复杂的例子，其中 $y = 0$ 与 $y = 1$ 两类仍无法用超平面分隔开，而一个可用的分类曲面可以表示为 $x_1^2 + x_2^2 = 0.5$。

对于此类非线性分类问题，直接使用前文介绍的 SVM 是难以处理的。解决的方法是将数据进行高维转换，将原问题转化为线性 SVM 可解的形式。为此，我们需要寻找一个映射函数，它能够对输入数据进行变换，将数据映射到高维空间中。同时，映射后的数据还需满足线性可分的性质，即在高维空间下能找到一个线性超平面，该超平面能够将正负样本正确划分。

假设输入空间为 $\mathcal{X}$，输出空间为 $\mathcal{Y}$，则映射函数可表示为

$$\phi(x) : \mathcal{X} \to \mathcal{Y}$$

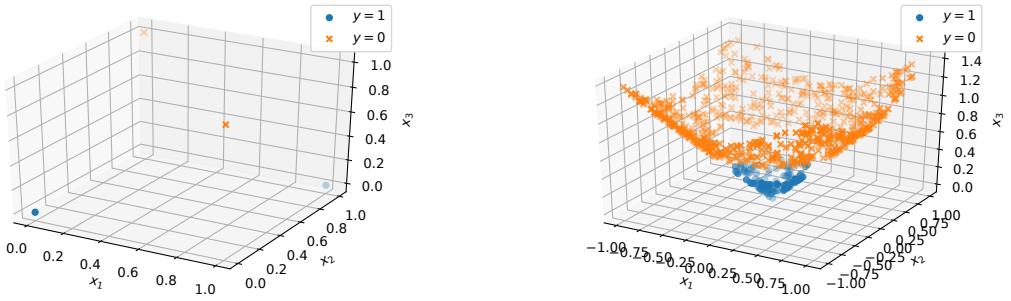
对于不同类型的数据，需根据数据的特点设置不同的映射函数，以实现高维空间下的线性可分。以上文介绍的二元异或函数为例，该问题的输入空间 $\mathcal{X} \subset \mathbf{R}^2$，若假设输出空间 $\mathcal{Y} \subset \mathbf{R}^3$，则一种可行的映射函数为

$$\phi(\boldsymbol{x}) = \left(x_1, x_2, (x_1 - x_2)^2\right) = \boldsymbol{y}.$$

可以看到，该函数将原数据映射到了三维空间中，如图3 (a) 所示。并且，通过这种映射，我们很容易能找到一个超平面（例如 $y_3 = 0.5$）将两类数据准确划分。由此看来，经过高维转换后，原来的非线性可分的问题被成功转化为新特征空间上的线性可分问题。

有了映射函数后，下一步可以使用 SVM 在新的特征空间下进行求解。以上文介绍的软间隔 SVM 为例，设样本 $\boldsymbol{x}$ 映射后的向量为 $\phi(\boldsymbol{x})$，将其代入软间隔 SVM 的对应方程中，可得到在新特征空间下的 SVM 方程。其中原始问题为

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



(a) 异或函数　　(b) 一个更复杂的例子

图 3: 映射后的线性不可分数据

其对偶问题为

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

SVM 分类界面公式为

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \phi(\boldsymbol{x}) + b = \sum_{i=1}^{N} \alpha_i y_i \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}) + b$$

至此，只需要使用 SMO 算法对新的对偶问题进行求解，就可以实现对非线性可分数据的分类。

1.6.2 正定核函数与 Mercer 定理

在上一小节中，我们已经通过高维转换，实现了使用软间隔 SVM 来求解非线性可分问题。这个过程主要分为两步：首先根据数据特征，寻找一个合适的映射函数 $\phi(x)$；其次使用软间隔 SVM 从映射后的数据中学习分类模型。这是否意味着非线性可分数据的 SVM 问题已经得到了完美解决呢？

细心的读者可能会发现，在该问题的求解中，一个重要的条件是数据的映射函数 $\phi(x)$ 是已知的，即我们需要显式地知道原始数据是经过了怎样的变换从而映射到新的特征空间，并在此基础上对映射后数据进行内积运算。而在实际应用中，映射后的特征空间一般都是高维的，而在高维特征空间中计算内积并不容易，且有可能出现“维数灾难”问题。例如，若输入数据维度为 2，即 $\boldsymbol{x} = (x_1, x_2)$，则最多能组合出 5 个不高于 2 次的项，即 $(x_1, x_2, x_1^2, x_2^2, x_1 x_2)$，因而可映射到 5 维空间中计算特征间的内积；而当输入数据是 3 维特征数据时，映射后空间可达 19 维；但如果原始数据达到 100 维甚至 1000 维时，映射后空间的维度是爆炸性增长的，此时计算数据间的内积将变得十分困难，甚至无法处理。

但注意到，在上述对偶问题和分类界面公式中，对于映射后数据，都只涉及到样本间的内积运算，即求解 $\phi(x)^T \phi(y)$，而对映射函数本身的形式并不关心。因此，是否能够对原问题进一步简化，找到一种方法能直接得到映射到高维空间后任意两个样本的内积结果，而绕过映射函数 $\phi(x)$ 的形式本身呢？核函数的引入恰恰解决了这个问题。

通常所说的核函数指的是正定核函数 (Positive-definite kernel function)。以下先给出正定核函数的基本定义。

定义 1.6.1 (正定核函数)**.** 设 $\mathcal{X} \subset \mathbf{R}^n$ 为输入空间, $\mathcal{H}$ 为希尔伯特空间, 若存在映射函数 $\phi : \mathcal{X} \to \mathcal{H}$, 使得二元函数 $K : \mathcal{X} \times \mathcal{X} \to \mathbf{R}$ 满足对任意 $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$，有

$$K(\boldsymbol{x}, \boldsymbol{z}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z}) \rangle = \phi(\boldsymbol{x})^T \phi(\boldsymbol{z}),$$

则称 $K(\boldsymbol{x}, \boldsymbol{z})$ 为正定核函数。

由此看来，核函数 K 像是一个“跳板”，让我们能够通过低维空间上的简单计算，得到高维空间上的内积结果。但是，根据以上定义，仍需先找到映射函数 $\phi(x)$ 的表达形式，再得到相应的核函数。那么，对于一般函数而言，能否跳过 $\phi(x)$，直接判断 K 是否为正定核函数呢？下面将给出正定核函数的另一种定义，亦可作为正定核函数的判断条件。

定义 1.6.2 (正定核函数的判定)**.** 若函数 $K : \mathcal{X} \times \mathcal{X} \to \mathbf{R}$ 满足以下两条性质

1. 对称性：对任意 $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$，有 $K(\boldsymbol{x}, \boldsymbol{z}) = K(\boldsymbol{z}, \boldsymbol{x})$。

2. 正定性：任取 N 个元素 $\boldsymbol{x}_i \in \mathcal{X}, i = 1, 2, ..., N$，函数 $K(\boldsymbol{x}, \boldsymbol{z})$ 对应的 Gram 矩阵是半正定矩阵。其中 Gram 矩阵为一个 $N \times N$ 的矩阵，第 (i, j) 个元素为 $K(\boldsymbol{x}_i, \boldsymbol{z}_j)$ 的值。

那么称 $K(\boldsymbol{x}, \boldsymbol{z})$ 为正定核函数。

接下来给出正定核函数判定条件的证明，即证明定理1.6.2与定理1.6.1是等价的。

证明. 先证明必要性，再证明充分性。

1. 必要性。已知存在映射函数 $\phi : \mathcal{X} \to \mathcal{H}$，使得对任意 $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$，有

$$K(\boldsymbol{x}, \boldsymbol{z}) = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z}) \rangle = \phi(\boldsymbol{x})^T \phi(\boldsymbol{z})$$

求证：函数 $K(\boldsymbol{x}, \boldsymbol{z})$ 满足对称性与正定性。

(a) 对称性。根据内积函数的对称性，对任意 $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$，有

$$\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{z}) \rangle = \langle \phi(\boldsymbol{z}), \phi(\boldsymbol{x}) \rangle$$

因此有 $K(\boldsymbol{x}, \boldsymbol{z}) = K(\boldsymbol{z}, \boldsymbol{x})$，对称性得证。

(b) 正定性。任取 N 个元素 $x_1, x_2, ..., x_N \in \mathcal{X}$，构建 $K(\boldsymbol{x}, \boldsymbol{z})$ 关于 $x_1, x_2, ..., x_N$ 的 Gram 矩阵

$$K = [K(x_i, x_j)]_{N \times N} = [K_{ij}]_{N \times N}$$

则只需证明：$\forall \alpha \in \mathbf{R}^N, \alpha^T K \alpha \geq 0.$

注意到，

$$\begin{aligned}
\alpha^T K\alpha &= \begin{pmatrix}\alpha_1 & \alpha_2 & ... & \alpha_N\end{pmatrix}\begin{pmatrix}K_{11} & K_{12} & ... & K_{1N}\\ K_{21} & K_{22} & ... & K_{2N}\\ ... & ... & & ...\\ K_{N1} & K_{N2} & ... & K_{NN}\end{pmatrix}\begin{pmatrix}\alpha_1\\ \alpha_2\\ ...\\ \alpha_N\end{pmatrix}\\
&= \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j K_{ij}\\
&= \sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j\phi(\boldsymbol{x}_i)^T\phi(\boldsymbol{x}_j)\\
&= \sum_{i=1}^{N}\alpha_i\phi(\boldsymbol{x_i})^T\sum_{j=1}^{N}\alpha_j\phi(\boldsymbol{x}_j)\\
&= \left[\sum_{i=1}^{N}\alpha_i\phi(\boldsymbol{x_i})\right]^T\left[\sum_{j=1}^{N}\alpha_j\phi(\boldsymbol{x}_j)\right]\\
&= \left\|\sum_{j=1}^{N}\alpha_j\phi(\boldsymbol{x}_j)\right\|^2 \geq 0
\end{aligned}$$

因此，$K(\boldsymbol{x},\boldsymbol{z})$ 关于 $x_1, x_2, ..., x_N$ 的 Gram 矩阵是半正定的。必要性得证。

2. 充分性。已知 $K:\mathcal{X}\times\mathcal{X}\to\mathbf{R}$ 是对称函数，且对任意 $x_1, x_2, ..., x_N\in\mathcal{X}$，$K(\boldsymbol{x},\boldsymbol{z})$ 关于 $x_1, x_2, ..., x_N$ 的 Gram 矩阵是半正定的。需证明：存在映射 $\phi:\mathcal{X}\to\mathcal{H}$，使得 $K(\boldsymbol{x},\boldsymbol{z})=\phi(\boldsymbol{x})^T\phi(\boldsymbol{z})$.

一般情况下，我们可假设 $\mathcal{X}\subset\mathbf{R}^n$，即输入空间是有限维的，这符合大多数数据的情况。在此情况下，Mercer 定理给出了充分性的结论 [1]。

定理 1.6.1 (Mercer 定理)**.** $\mathbf{R}^N\times\mathbf{R}^N\to\mathbf{R}$ 上的映射 $K(\boldsymbol{x},\boldsymbol{z})$ 是一个有效核函数（也称 Mercer 核函数）当且仅当对于训练样本其相应的核函数 Gram 矩阵是对称半正定的，即对于任何平方可积函数 $g(\boldsymbol{x})$ 有 $\int\int g(\boldsymbol{x})K(\boldsymbol{x},\boldsymbol{z})g(\boldsymbol{y})d\boldsymbol{x}d\boldsymbol{y}\geq 0$（对于离散情况，则需满足对于任意 $\alpha\in\mathbf{R}^N$，都有 $\alpha^T K\alpha\geq 0$，其中 K 为核函数在有限数据集下的 Gram 矩阵）。

Mercer 定理指出了判断正定核函数的一种特殊情况，即当输入空间是 $\mathbf{R}^n$ 的子集时，只需要通过验证函数 K 在有限训练样本上的 Gram 矩阵是否为半正定矩阵，即可判断该函数是否为正定核函数。在大多数情况下，样本的输入空间都是有限维的，因此可用此定理进行判断。而对于更一般的情况，则需要借助再生希尔伯特空间等概念来对充分性进行证明，文献 [9] 给出了详细的证明内容，读者可自行参考。

■

至此，我们便可使用核函数将 SVM 的对偶问题中的内积替换为核函数，达到跳过映射函数，简化计

算的目的。使用了核函数的支持向量机也被称为核 SVM，其对偶形式为

$$\begin{aligned} \arg\max_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^{N} a_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i, \boldsymbol{x}_j) \\ \text{s.t.} \quad & \sum_{i=1}^{N}\alpha_i y_i = 0, \\ & 0 \le \alpha_i \le C, i = 1, 2, \ldots, N. \end{aligned} \tag{77}$$

对应的分类界面公式为

$$f(\boldsymbol{x}) = \sum_{i=1}^{N}\alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b \tag{78}$$

当 $K(\boldsymbol{x}, \boldsymbol{z})$ 为正定核函数时，Eq.(77)为凸二次规划问题，亦可使用 SMO 算法进行求解。

1.6.3 常用核函数

根据定义1.6.1，我们可以构造出符合条件的核函数，但判定某个函数是否为核函数却并不容易，尤其是当训练数据集样本量大或维度较高时。在实际应用中，根据数据特性的不同，往往直接使用已得到证明的正定核函数作为 SVM 的核函数，以下列出了一些常用的核函数供参考。

1. 线性核函数 (Linear Kernel Function)

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

线性核函数即为基本的内积函数，并不对原始数据进行映射，采用此核函数的核 SVM 与前文所说的线性 SVM 没有区别。其优点在于参数少，速度快，对于线性可分的数据有较好的效果。同时，对于维度较高的数据，也往往优先选择线性核函数，以保证计算耗时在可接受的水平内。

2. 多项式核函数 (Polynomial Kernel Function)

$$K(\boldsymbol{x}, \boldsymbol{z}) = (\gamma\boldsymbol{x}^T\boldsymbol{z} + r)^p, \gamma > 0.$$

多项式核函数包含三个参数 γ, r, p，对应了一个 p 次多项式支持向量机，适用于线性不可分数据的学习。但其缺点在于参数较多，增加了模型调试的复杂性；且随着多项式阶数的升高，映射维度增大，核函数 Gram 矩阵中的值将趋近无穷大或无穷小，导致计算困难，也容易产生“过拟合”现象。

3. 径向基核函数 (Radial Basis Function Kernel，RBF 核)

$$K(\boldsymbol{x}, \boldsymbol{z}) = \exp\left(-\gamma\|\boldsymbol{x} - \boldsymbol{z}\|^2\right), \gamma > 0.$$

RBF 核是非线性 SVM 中应用最为广泛的的核函数，对于小样本与大样本均有不错的效果。同时，文献 [5] 指出，线性核函数可以认为是 RBF 核的一种特殊情况，若在核函数选择中使用了 RBF 核，则无需考虑线性核的情况，但计算时间相比线性核较长。与多项式核函数相比，RBF 核需要确定的参数较少，在一定程度上减轻了参数调试的负担。

4. Sigmoid 核函数

$$K(\boldsymbol{x}, \boldsymbol{z}) = \tanh(\gamma\boldsymbol{x}^T\boldsymbol{z} + r).$$

Sigmoid 核函数也是非线性 SVM 中常用的核函数之一，其中 γ, r 为需要调整的参数。

在实际应用中，可根据专家先验知识预先选定核函数，或是使用交叉验证方法尝试不同核函数及其参数选择，找出归纳误差最小的核函数及其参数组合使用 [4]。除此之外，文献 [8] 给出了一种混合核函数方法，通过将不同的核函数组合起来，达到更好的分类效果。

1.7 序列最小优化算法

如前文所述，支持向量机的训练过程可表示为求解具有最优解的凸二次规划问题，许多通用算法均可求解此类问题，但当训练数据集样本量较大时会带来很大的开销，导致效率较低，甚至无法使用。为了缓解这个问题，许多高效的 SVM 求解方法被提出，而目前应用最为广泛的是序列最小优化算法。本节将对该算法的运行流程进行详细讨论，并给出该算法的推导过程。

序列最小优化算法 (Sequential Minimal Optimization, SMO) 由 Platt 于 1998 年提出，是一种启发式学习算法 [6]。SMO 算法的主要思想是将一个复杂的二次规划问题分解为多个只有两个变量的二次规划子问题，并通过求解子问题来解决原始问题。算法将检查当前是否仍存在未满足 KKT 条件的变量，若有，则选取两个变量（至少有一个不满足 KKT 条件）作为当前问题的待优化目标，固定其他变量，在此基础上求得优化目标的解析解。当所有变量的 KKT 条件均得到满足，原问题的就得到了。

1.7.1 算法运行流程

根据上文所述，带软间隔的核 SVM 优化目标函数为 Eq.(77)。通过改变符号，可将求极大值问题转化为求极小值问题，于是原问题等价于求解

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(\boldsymbol{x}_i,\boldsymbol{x}_j) - \sum_{i=1}^{N}\alpha_i \tag{79}$$

$$\text{s.t.} \quad \sum_{i=1}^{N}\alpha_i y_i = 0, \tag{80}$$

$$0 \le \alpha_i \le C, i = 1,2,\ldots,N. \tag{81}$$

其中，待优化目标为拉格朗日乘子 $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_N)$，$N$ 为训练数据集样本数目，每个乘子 α_i 对应了训练集上的一个样本点 $(\boldsymbol{x}_i, y_i)$。

记 $f(\boldsymbol{x}_i) = \sum_{j=1}^{N}\alpha_j y_j K(\boldsymbol{x}_j, \boldsymbol{x}_i) + b$ 表示 SVM 在样本 $\boldsymbol{x}_i$ 下的预测结果，$K(\boldsymbol{x}_i, \boldsymbol{z}_i)$ 为核函数，且记 $K(\boldsymbol{x}_i, \boldsymbol{z}_i) = K_{ij}$。SMO 算法的总体流程如下

输入：训练数据集 $T = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), ..., (\boldsymbol{x}_N, y_n)\}$，其中 $\boldsymbol{x}_i \in X = \boldsymbol{R}^m, y_i \in Y = \{-1, 1\}, i = 1, 2, ..., N$；设定精度为 ε;

输出：SVM 的近似解 $(\widehat{\boldsymbol{\alpha}}, \widehat{b})$;

1. 算法初始化：令 $k = 0, \boldsymbol{\alpha}^{(0)} = 0$;

2. 使用启发式方法选取待优化变量 $\alpha_1^{(k)}, \alpha_2^{(k)}$:

   (a) 计算误差 $E_i^{(k)} = f(\boldsymbol{x}_i) - y_i, i = 1, 2, ..., N$;

   (b) 对每个 $\alpha_i^{(k)}, i = 1, 2, ..., N$，检查是否满足本问题的 KKT 条件，即 Eq.(91)～Eq.(93);

   (c) 选取一个不满足 KKT 条件的变量，记为 $\alpha_2^{(k)}$;

   (d) 从 $\alpha_2^{(k)}$ 以外的变量中选取 $\alpha_1^{(k)}$，使得 $|E_1^{(k)} - E_2^{(k)}|$ 最大。

3. 使用解析法计算 $\alpha_1^{(k)}, \alpha_2^{(k)}$ 的最优解并更新至 $\alpha_1^{(k+1)}, \alpha_2^{(k+1)}$，$\boldsymbol{\alpha}$ 中的其他变量不变，得到 $\boldsymbol{\alpha}^{(k+1)}, b^{(k+1)}$：

(a) 计算 $\alpha_2^{(k+1)}$ 的取值范围 $[L, H]$，其中：

$$L = \begin{cases} \max(0, \alpha_2^{(k)} - \alpha_1^{(k)}), & \text{if } y_1 \neq y_2 \\ \max(0, \alpha_2^{(k)} + \alpha_1^{(k)} - C), & \text{if } y_1 = y_2 \end{cases} \tag{82}$$

$$H = \begin{cases} \min(C, C + \alpha_2^{(k)} - \alpha_1^{(k)}), & \text{if } y_1 \neq y_2 \\ \min(C, \alpha_2^{(k)} + \alpha_1^{(k)}), & \text{if } y_1 = y_2 \end{cases} \tag{83}$$

(b) 求得最优解 $\alpha_1^{(k+1)}, \alpha_2^{(k+1)}, b^{(k+1)}$：

首先根据解析法求得原问题的最优解

$$\alpha_2^* = \alpha_2^{(k)} + \frac{y_2(E_1 - E_2)}{K_{11} + K_{22} - 2K_{12}}, \tag{84}$$

结合 $\alpha_2^{(k+1)}$ 的取值范围，可得

$$\alpha_2^{(k+1)} = \begin{cases} H, & \alpha_2^* > H \\ \alpha_2^*, & L \leq \alpha_2^* \leq H \\ L, & \alpha_2^* < L \end{cases} \tag{85}$$

由 $\alpha_2^{(k+1)}$ 可求得 $\alpha_1^{(k+1)}$，

$$\alpha_1^{(k+1)} = \alpha_1^{(k)} + y_1 y_2 (\alpha_2^{(k)} - \alpha_2^{k+1}). \tag{86}$$

之后更新参数 b

$$b_1^* = y_1 - \alpha_1^{(k+1)} y_1 K_{11} - \alpha_2^{(k+1)} y_2 K_{12} - \sum_{i=3}^{N} \alpha_i^{(k)} y_i K_{1i}, \tag{87}$$

$$b_2^* = y_2 - \alpha_1^{(k+1)} y_1 K_{21} - \alpha_2^{(k+1)} y_2 K_{22} - \sum_{i=3}^{N} \alpha_i^{(k)} y_i K_{2i}, \tag{88}$$

$$b^{(k+1)} = \begin{cases} b_1^*, & \alpha_1^{(k+1)} \in (0, C) \text{ and } \alpha_2^{(k+1)} \in (0, C) \\ \dfrac{b_1^* + b_2^*}{2}, & \text{others.} \end{cases} \tag{89}$$

(c) 更新 E_i 值

$$E_i^{(k+1)} = \sum_{j=1}^{N} y_j \alpha_j^{(k+1)} K_{ij} + b^{(k+1)} - y_i. \tag{90}$$

4. 检查在精度 ε 范围内，是否满足以下停止条件（KKT 条件）

$$\sum_{i=1}^{N} \alpha_i y_i = 0, \tag{91}$$

$$0 \leq \alpha_i \leq C,\ i = 1, 2, ..., N, \tag{92}$$

$$y_i f(\boldsymbol{x}_i) \begin{cases} \geq 1, & i \in \{i|\alpha_i = 0\} \\ = 1, & i \in \{i|0 < \alpha_i < C\} \\ \leq 1, & i \in \{i|\alpha_i = C\} \end{cases} \tag{93}$$

若满足停止条件，则算法终止；否则令 $k = k + 1$，转步骤 2.

■

1.7.2 SMO 算法的推导

以下将介绍 SMO 算法的具体推导过程。除特殊注明外，其余数学符号定义与上节相同。

二元凸优化问题的闭式解 假设在某次迭代时，被选中的两个变量为 α_1, α_2，其他变量固定。则对于只包含两个变量 α_1, α_2 的二次规划问题，Eq.(79)可整理为以下形式

$$\begin{aligned} W(\alpha_1, \alpha_2) =& \frac{1}{2}K_{11}\alpha_1^2 + \frac{1}{2}K_{22}\alpha_2^2 + K_{12}y_1y_2\alpha_1\alpha_2 - (\alpha_1 + \alpha_2) \\ &+ y_1\alpha_1\sum_{i=3}^{N} y_i\alpha_i K_{1i} + y_2\alpha_2\sum_{i=3}^{N} y_i\alpha_i K_{2i} + c, \end{aligned} \tag{94}$$

其中，$K_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$，$c$ 表示与 α_1, α_2 无关的常量。

根据第一个 KKT 条件 $\sum_i \alpha_i y_i = 0$，有

$$\alpha_1 y_1 + \alpha_2 y_2 = -\sum_{i=3}^{N} \alpha_i y_i = \varsigma \tag{95}$$

又因为 y_i 为表示样本 i 的真实分类，$y_i \in \{-1, 1\}$，因此有 $y_i^2 = 1$，故

$$\alpha_1 = (\varsigma - \alpha_2 y_2)y_1 \tag{96}$$

代入 Eq.(94)，得到

$$\begin{aligned} W(\alpha_2) =& \frac{1}{2}K_{11}(\varsigma - y_2\alpha_2)^2 + \frac{1}{2}K_{22}\alpha_2^2 + (\varsigma - y_2\alpha_2)K_{12}y_2\alpha_2 - [(\varsigma - \alpha_2 y_2)y_1 + \alpha_2] \\ &+ (\varsigma - y_2\alpha_2)\sum_{j=3}^{N} y_j\alpha_j K_{1j} + y_2\alpha_2\sum_{j=3}^{N} y_j\alpha_j K_{2j} + c \end{aligned} \tag{97}$$

为简便起见，记

$$v_i = \sum_{j=3}^{N} y_j\alpha_j K_{ij} = f(\boldsymbol{x}_i) - y_1\alpha_1^{old}K_{i1} - y_2\alpha_2^{old}K_{i2} - b^{old}, i = 1, 2,$$

其中 $\alpha_1^{old}, \alpha_2^{old}, b^{old}$ 为对应参数在上一次迭代时的值，$f(\boldsymbol{x}_i)$ 为上一次迭代后 SVM 在样本 $\boldsymbol{x}_i$ 下的预测结果，因此 v_1, v_2 都是常量。将 v_i 定义代入 Eq.(97) 并化简可以得到

$$\begin{aligned} W(\alpha_2) =& \frac{1}{2}K_{11}(\varsigma - y_2\alpha_2)^2 + \frac{1}{2}K_{22}\alpha_2^2 + (\varsigma - y_2\alpha_2)K_{12}y_2\alpha_2 \\ &- [(\varsigma - \alpha_2 y_2)y_1 + \alpha_2] + (\varsigma - y_2\alpha_2)v_1 + y_2\alpha_2 v_2 + c. \end{aligned} \tag{98}$$

容易想到，通过对目标函数进行求导，可以得到该极小值问题的解。令

$$\frac{\partial W}{\partial \alpha_2} = -K_{11}(\varsigma - y_2\alpha_2)y_2 + K_{22}\alpha_2 + \varsigma K_{12}y_2 - 2K_{12}\alpha_2 - [-y_1y_2 + 1] - y_2v_1 + y_2v_2 = 0,$$

有

$$\begin{aligned}(K_{11} + K_{22} - 2K_{12})\alpha_2 &= K_{11}\varsigma y_2 - K_{12}\varsigma y_2 - y_1y_2 + 1 + y_2v_1 - y_2v_2 \\ &= y_2(K_{11}\varsigma - K_{12}\varsigma - y_1 + y_2 + v_1 - v_2)\end{aligned} \tag{99}$$

将 $\varsigma = y_1\alpha_1^{old} + y_2\alpha_2^{old}$ 及 v_1, v_2 的定义代入上式并整理，可得到 α_2 解析解，即

$$\alpha_2^{\star} = \alpha_2^{old} + \frac{y_2(E_2 - E_1)}{K_{11} + K_{22} - 2K_{12}}. \tag{100}$$

确定参数边界范围 在更新参数前，我们还需要解决一个重要问题，即确定 α_2 的边界约束。首先，注意到原问题给出的约束条件 $0 \le \alpha_i \le C, i \in 1, 2$，则在以 α_2 为纵轴，α_1 为横轴的二维直角坐标系上，点 (α_1, α_2) 应落在 $[0, C] \times [0, C]$ 的矩形区域中，即参数 α_1, α_2 应该满足以下条件

$$\begin{cases}\alpha_1 \ge 0, \\ \alpha_2 \ge 0, \\ \alpha_1 \le C, \\ \alpha_2 \le C.\end{cases} \tag{101}$$

更进一步，由 Eq.(95)，我们知道 (α_1, α_2) 应落在一条直线上，而通过联立该直线与 Eq.(101)围成的约束区域，我们可以很容易得到 α_2 的取值范围。注意到 y_i 的取值只可能为 -1 或 $+1$，因此该直线的斜率绝对值为 1，根据 y_1, y_2 的取值不同，需进行分类讨论：

1. 若 $y_1 \ne y_2$，则 Eq.(95) 可化为 $\alpha_2 - \alpha_1 = k$，其中 k 为常数。将该直线与 Eq.(101) 中描述的约束区联立，消去 α_1，可得到关于 α_2 的约束条件，其中

$$\begin{cases}\alpha_2 \ge k, \\ \alpha_2 \le C + k,\end{cases} \tag{102}$$

因此有

$$\max(0, \alpha_2^{old} - \alpha_1^{old}) \le \alpha_2^{new} \le \min(C, C + \alpha_2^{old} - \alpha_1^{old})$$

2. 若 $y_1 = y_2$，则 Eq.(95) 可化为 $\alpha_2 + \alpha_1 = k$，其中 k 为常数。同样将该直线与上述约束条件联立，可得到关于 α_2 的约束条件，即

$$\begin{cases}\alpha_2 \le k, \\ \alpha_2 \ge k - C,\end{cases} \tag{103}$$

因此有

$$\max(0, \alpha_2^{old} + \alpha_1^{old} - C) \le \alpha_2^{new} \le \min(C, \alpha_2^{old} + \alpha_1^{old}).$$

综合上述两种情况，可得到参数 α_2 的约束边界 $[L, H]$，即

$$L = \begin{cases}\max(0, \alpha_2^{old} - \alpha_1^{old}), & \text{if } y_1 \ne y_2 \\ \max(0, \alpha_2^{old} + \alpha_1^{old} - C), & \text{if } y_1 = y_2\end{cases}$$

$$H = \begin{cases} \min(C, C + \alpha_2^{old} - \alpha_1^{old}), & \text{if } y_1 \neq y_2 \\ \min(C, \alpha_2^{old} + \alpha_1^{old}), & \text{if } y_1 = y_2 \end{cases}$$

因此，参数 α_2 的更新值应为考虑了约束边界后的结果，即

$$\alpha_2^{new} = \begin{cases} H, & \alpha_2^\star > H \\ \alpha_2^\star, & L \leq \alpha_2^\star \leq H \\ L, & \alpha_2^\star < L \end{cases} \tag{104}$$

在此基础上，我们可以得到 α_1 的解，根据 Eq.(95) 可知必成立

$$\alpha_1^{new} y_1 + \alpha_2^{new} y_2 = \alpha_1^{old} y_1 + \alpha_2^{old} y_2,$$

因此

$$\alpha_1^{new} = \alpha_1^{old} + y_1 y_2 (\alpha_2^{old} - \alpha_2^{new}). \tag{105}$$

确定截距参数 下一步，需要求出参数 b 的值。根据 KKT 条件 Eq.(93)，当 $0 < \alpha_1^{(k+1)} < C$ 时，有

$$y_1 f(\boldsymbol{x}_1) = 1,$$

即 $f(\boldsymbol{x}_1) = y_1$，将该式展开可得

$$\alpha_1^{new} y_1 K_{11} + \alpha_2^{new} y_2 K_{12} + \sum_{i=3}^{N} \alpha_i y_i K_{1i} + b = y_1,$$

因此其对应的参数 b 为

$$b_1^* = y_1 - \alpha_1^{new} y_1 K_{11} - \alpha_2^{new} y_2 K_{12} - \sum_{i=3}^{N} \alpha_i y_i K_{1i}.$$

同理，当 $0 < \alpha_2^{(k+1)} < C$ 时，其对应的参数 b 为

$$b_2^* = y_2 - \alpha_1^{new} y_1 K_{21} - \alpha_2^{new} y_2 K_{22} - \sum_{i=3}^{N} \alpha_i y_i K_{2i}.$$

在每一轮迭代时，均需要对参数 b 进行更新。当 $0 < \alpha_1^{new} < C$ 与 $0 < \alpha_2^{new} < C$ 同时满足时，$b_1^* = b_2^*$，此时可更新 $b^{new} = b_1^*$；否则，b_1^* 与 b_2^* 围成的区间均符合 KKT 条件，此时将取区间中点作为 b^{new} 的更新值 [6]。■

至此，SMO 算法的推导已基本完成。简单总结一下，SMO 算法的主要思想是将一个难解的大型凸优化问题分解成一系列子问题，并且这些子问题可通过解析法直接得到最优解，省去了子问题的数值优化过程。通过判断待优化变量在 KKT 条件上的满足情况，可以快速找到每轮迭代的子问题。当所有变量的 KKT 条件均得到满足时，原问题也得到解决。同时，SMO 算法的内存需求是随训练数据集样本数而线性增长的，且算法中没有矩阵计算的过程，这使得其在大数据集上有较好的可扩展性。有关 SMO 算法的更多细节，读者可阅读文献 [6] 做进一步了解。

1.8 SVM 的回归形式

本节介绍支持向量机在回归问题上的应用。给定数据集 $T = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$，其标签 $y_i \in \mathbb{R}$ 为连续值，回归问题的目标是寻找一个函数 $f(\boldsymbol{x})$，使得对于任意样本 $(\boldsymbol{x}_i, y_i)$，模型的预测结果 $f(\boldsymbol{x}_i)$ 与真实值 y_i 尽可能接近。传统的线性回归方法是通过最小化样本的平方损失，即 $\sum_i (f(\boldsymbol{x}_i) - y_i)^2$ 来对模型参数进行估计，但在 SVM 中并不这样做。

支持向量回归 (Support Vector Regression, SVR) 是 SVM 在回归问题上的扩展 [3]。我们希望将回归问题表示为形如 Eq.(47)的形式。在分类问题中，SVM 的约束条件是每个样本根据其所属类别标签落在分类超平面的对应一侧，但在回归问题中，数据标签均为连续值，不存在"类别"的概念，因而无法直接沿用分类 SVM 的约束形式。针对这个问题，SVR 采用了一种基于回归误差的 $\epsilon-$ 不敏感性约束作为其约束条件。具体来说，记回归函数为

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

其中 $\boldsymbol{w}, b$ 为待优化变量，则 SVR 要求在给定 $\epsilon \geq 0$ 时，有

$$|f(\boldsymbol{x}_i) - y_i| \leq \epsilon, \tag{106}$$

对所有的 $i = 1, 2, ..., N$ 均成立。因此，SVR 的约束条件允许预测误差在 $[-\epsilon, \epsilon]$ 范围内（也称 $\epsilon-$ 不敏感区间），这与传统回归算法不同。

在此基础上，可引入松弛变量，允许模型在一定程度上的预测错误。注意到 Eq.(106) 中的绝对值函数，可知应允许两个方向上不同大小的松弛操作，因此应该存在两组松弛变量。基于此，我们可导出 SVR 的目标函数，即

$$\min_{\boldsymbol{w}, b, \boldsymbol{\xi}^u, \boldsymbol{\xi}^v} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^N (\xi_i^u + \xi_i^v) \tag{107}$$

$$\text{s.t.} \quad f(\boldsymbol{x}_i) - y_i \leq \epsilon + \xi_i^v, \tag{108}$$

$$y_i - f(\boldsymbol{x}_i) \leq \epsilon + \xi_i^u, \tag{109}$$

$$\xi_i^u \geq 0, \xi_i^v \geq 0, \quad i = 1, 2, \ldots, N. \tag{110}$$

图4展示了 SVR 的约束条件，其中，绿色区间为 SVR 允许的 ϵ 误差间隔，当预测样本落入此区间内时可认为预测是正确的；黄色区间为某两个样本对应的软间隔松弛变量，可知松弛变量能够在两个方向进行松弛。在理想状态下，当训练结束时，所有样本应落入绿色区间或黄色区间中，即原问题的约束条件得到满足。

接下来需要对 SVR 的目标函数进行求解。与分类 SVM 相同，依然使用拉格朗日乘子法构造目标函数，得到

$$\begin{aligned} L(\boldsymbol{w}, b, \boldsymbol{\xi}^u, \boldsymbol{\xi}^v, \boldsymbol{\alpha}^u, \boldsymbol{\alpha}^v, \boldsymbol{\delta}^u, \boldsymbol{\delta}^v) =& \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^N (\xi_i^u + \xi_i^v) + \sum_{i=1}^N \alpha_i^v (f(\boldsymbol{x_i}) - y_i - \epsilon - \xi_i^v) \\ &+ \sum_{i=1}^N \alpha_i^u (y_i - f(\boldsymbol{x}_i) - \epsilon - \xi_i^u) - \sum_{i=1}^N \delta_i^u \xi_i^u - \sum_{i=1}^N \delta_i^v \xi_i^v, \end{aligned} \tag{111}$$

其中，$\alpha_i^u \geq 0, \alpha_i^v \geq 0, \delta_i^u \geq 0, \delta_i^v \geq 0$。原问题的无约束形式为

$$\min_{\boldsymbol{w}, b, \boldsymbol{\xi}^u, \boldsymbol{\xi}^v} \max_{\alpha_i^u \geq 0, \alpha_i^v \geq 0, \delta_i^u \geq 0, \delta_i^v \geq 0} L(\boldsymbol{w}, b, \boldsymbol{\xi}^u, \boldsymbol{\xi}^v, \boldsymbol{\alpha}^u, \boldsymbol{\alpha}^v, \boldsymbol{\delta}^u, \boldsymbol{\delta}^v) = p^\star.$$

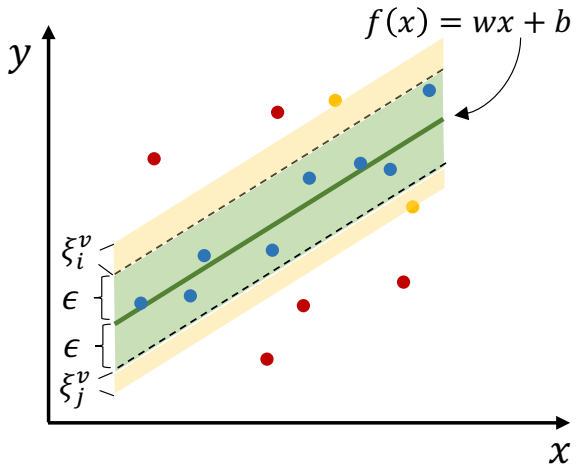


图 4: SVR 约束条件示意图

进一步，我们将其转化为对偶问题：

$$\max_{\alpha_i^u \geq 0, \alpha_i^v \geq 0, \delta_i^u \geq 0, \delta_i^v \geq 0} \min_{\boldsymbol{w}, b, \boldsymbol{\xi}^u, \boldsymbol{\xi}^v} L(\boldsymbol{w}, b, \boldsymbol{\xi}^u, \boldsymbol{\xi}^v, \boldsymbol{\alpha}^u, \boldsymbol{\alpha}^v, \boldsymbol{\delta}^u, \boldsymbol{\delta}^v) = d^\star.$$

容易验证，在该问题中 KKT 条件是成立的。因此，我们让 L 对于 $\boldsymbol{w}, b, \xi_i^u, \xi_i^v$ 分别求偏导，并令其为 0，可得

$$\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{w}} = 0 \Rightarrow \boldsymbol{w} &= \sum_{i=1}^{N} (\alpha_i^u - \alpha_i^v) \boldsymbol{x}_i, \qquad (112)\\
\frac{\partial L}{\partial b} = 0 \Rightarrow \sum_{i=1}^{N} (\alpha_i^u - \alpha_i^v) &= 0,\\
\frac{\partial L}{\partial \xi_i^u} = 0 \Rightarrow C &= \alpha_i^u + \delta_i^u,\\
\frac{\partial L}{\partial \xi_i^v} = 0 \Rightarrow C &= \alpha_i^v + \delta_i^v.
\end{aligned}$$

将上述结果回带到函数 L 中并化简，可导出对偶问题外层优化的目标函数：

$$\max_{\boldsymbol{\alpha}^u, \boldsymbol{\alpha}^v} \quad \sum_{i=1}^{N} [(y_i - \epsilon)\alpha_i^u - (y_i + \epsilon)\alpha_i^v] - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\alpha_i^u - \alpha_i^v)(\alpha_j^u - \alpha_j^v) \boldsymbol{x}_i^T \boldsymbol{x}_j \qquad (113)$$

$$\text{s.t.} \quad \sum_{i=1}^{N} (\alpha_i^u - \alpha_i^v) = 0, \qquad (114)$$

$$0 \leq \alpha_i^u \leq C, 0 \leq \alpha_i^v \leq C, i = 1, 2, \ldots, N. \qquad (115)$$

同样，该优化问题也可用上一节中介绍的 SMO 算法进行求解。

那么，SVR 问题中的支持向量应该如何定义呢？根据 Eq.(112)中的结果可知，当仅当 $\alpha_i^u - \alpha_i^v \neq 0$ 时，样本 $(\boldsymbol{x}_i, y_i)$ 对应的权重 $w_i \neq 0$，这样的样本就是 SVR 中的支持向量。考虑 KKT 条件中的互补松弛项，即

$$\begin{aligned}
\alpha_i^u (y_i - f(\boldsymbol{x}_i) - \epsilon - \xi_i^u) &= 0\\
\alpha_i^v (f(\boldsymbol{x}_i) - y_i - \epsilon - \xi_i^v) &= 0
\end{aligned}$$

因此存在如下关系

$$\begin{aligned} y_i - f(\boldsymbol{x}_i) - \epsilon - \xi_i^u \neq 0 \Rightarrow \quad \alpha_i^u = 0, \\ y_i - f(\boldsymbol{x}_i) - \epsilon - \xi_i^u = 0 \Rightarrow \quad \alpha_i^u \neq 0, \\ f(\boldsymbol{x}_i) - y_i - \epsilon - \xi_i^v \neq 0 \Rightarrow \quad \alpha_i^v = 0, \\ f(\boldsymbol{x}_i) - y_i - \epsilon - \xi_i^v = 0 \Rightarrow \quad \alpha_i^v \neq 0. \end{aligned}$$

若样本 $(\boldsymbol{x}_i, y_i)$ 落在 $\epsilon-$ 不敏感区间内，即满足 $|f(\boldsymbol{x}_i) - y_i)| \leq \epsilon$ 时，两个松弛变量 $\xi_i^u = \xi_i^v = 0$，且必有 $y_i - f(\boldsymbol{x}_i) - \epsilon - \xi_i^u \neq 0$ 以及 $f(\boldsymbol{x}_i) - y_i - \epsilon - \xi_i^v \neq 0$，这时 $\alpha_i^u = \alpha_i^v = 0$，该样本必然不是支持向量。因此，SVR 的支持向量一定落在 $\epsilon-$ 不敏感区间外。

最终，SVR 的预测函数为

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} (\alpha_i^u - \alpha_i^v) \boldsymbol{x}_i^T \boldsymbol{x} + b,$$

若考虑核方法，可同理推出核 SVR 的预测函数

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} (\alpha_i^u - \alpha_i^v) K(\boldsymbol{x}_i, \boldsymbol{x}) + b,$$

其中 $K(\boldsymbol{x}, \boldsymbol{z})$ 为核函数。

2 Relevance Vector Machine

相关向量机 (Relevance Vector Machine, RVM) 由 Tipping 于 2001 年提出，是一种基于贝叶斯框架的预测模型 [7]。RVM 使用了与 SVM 相似的函数形式，但将贝叶斯学习框架嵌入其中，使预测结果概率化。SVM 与 RVM 都属于基于核的稀疏解算法，即它们都只依赖数据集中一个子集即可进行预测，这个子集在 SVM 中即为前文所述的支持向量，而在 RVM 中被称为相关向量 (Relevance Vector)。与 SVM 相比，RVM 通过引入概率模型，带来了以下好处 [7]:

1. 提供了预测的后验概率。传统 SVM 模型只能提供二元的分类结果，而 RVM 使用的概率模型提供了预测结果的概率分布，这使我们能够得到更多的信息，并给不确定性分析带来了可能性。

2. 降低了噪音带来的敏感性。在 SVM 中，我们使用软间隔来消除数据中噪声的影响，但软间隔带有一个超参数 C 需要手动调整。而实践证明，惩罚系数 C 的选择对模型的预测性能有较大的影响，往往需要通过交叉验证来选择合理的 C 值。RVM 通过使用概率模型来解释数据中的噪声，免去了手工调整惩罚因子 C 的过程。

3. 提供更加稀疏的解。RVM 输出的相关向量数目要少于 SVM 的支持向量数目，这也意味着 RVM 能提供更快的预测速度。

4. RVM 不再要求核函数是正定的。

本章将对 RVM 进行简要介绍。

2.1 RVM 的目标函数

假设训练数据集 $T = \{(\boldsymbol{x}_1, t_1), (\boldsymbol{x}_2, t_2), ..., (\boldsymbol{x}_N, t_n)\}$，其中 $\boldsymbol{x}_i \in \boldsymbol{X} \subset \boldsymbol{R}^m$ 为数据特征，t_i 为待预测标签，$i = 1, 2, ..., N$。首先回忆一下 SVM 的情况。假设模型参数为 $\boldsymbol{w}$，$K(\boldsymbol{x}, \boldsymbol{z})$ 为核函数，在 SVM 中，我们使用

$$y(\boldsymbol{x}; \boldsymbol{w}) = \sum_{i=1}^{N} w_i K(\boldsymbol{x}_i, \boldsymbol{x}) + w_0 \tag{116}$$

作为算法对于输入样本 $\boldsymbol{x}$ 对应标签 t 的预测，即

$$\hat{t} = y(\boldsymbol{x}; \boldsymbol{w}).$$

在 RVM 中，同样使用 Eq.(116) 作为 t 的预测值参考，不同之处在于 RVM 使用了概率的视角看待问题。具体来说，对于任一样本 $(\boldsymbol{x}_i, t_i)$，均假设

$$t_i \sim N(y(\boldsymbol{x}_i; \boldsymbol{w}), \sigma^2)$$

即 t_i 服从以 $y(\boldsymbol{x}_i, \boldsymbol{w})$ 为均值，σ^2 为方差的正态分布。因此其概率密度函数为

$$f(t_i; y(\boldsymbol{x}_i; \boldsymbol{w}), \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(t_i - y(\boldsymbol{x}_i; \boldsymbol{w}))^2}{2\sigma^2}).$$

同时，假设所有样本是相互独立的，则我们可以得到在给定 σ^2 和模型参数 $\boldsymbol{w}$ 下，$\boldsymbol{t}$ 的条件概率分布，即

$$\begin{aligned} p(\boldsymbol{t}|\boldsymbol{w},\sigma^2) &= \prod_{i=1}^{N} f(t_i; f(\boldsymbol{x}_i;\boldsymbol{w}),\sigma^2) \\ &= \prod_{i=1}^{N} \left[(2\pi\sigma^2)^{\frac{1}{2}} \exp(-\frac{(t_i - y(\boldsymbol{x}_i;\boldsymbol{w}))^2}{2\sigma^2}) \right] \\ &= (2\pi\sigma^2)^{-\frac{N}{2}} \exp(-\frac{\|\boldsymbol{t} - \Phi\boldsymbol{w}\|^2}{2\sigma^2}) \end{aligned} \tag{117}$$

其中 $\boldsymbol{w}$ 为包含了偏置 w_0 的权重列向量 $(w_0, w_1, ..., w_N)^T$，Φ 为增加了偏置列后核函数 $K(\boldsymbol{x}, \boldsymbol{z})$ 的 Gram 矩阵，即

$$\Phi = \begin{pmatrix} 1 & K(\boldsymbol{x}_1,\boldsymbol{x}_1) & K(\boldsymbol{x}_1,\boldsymbol{x}_2) & ... & K(\boldsymbol{x}_1,\boldsymbol{x}_N) \\ 1 & K(\boldsymbol{x}_2,\boldsymbol{x}_1) & K(\boldsymbol{x}_2,\boldsymbol{x}_2) & ... & K(\boldsymbol{x}_2,\boldsymbol{x}_N) \\ ... & ... & ... & ... & ... \\ 1 & K(\boldsymbol{x}_N,\boldsymbol{x}_1) & K(\boldsymbol{x}_N,\boldsymbol{x}_2) & ... & K(\boldsymbol{x}_N,\boldsymbol{x}_N) \end{pmatrix}.$$

若直接对 Eq.(117) 使用极大似然法估计 $\boldsymbol{w}, \sigma^2$，则会产生较为严重的过拟合现象，且权重 $\boldsymbol{w}$ 中的大部分元素将不为 0，也即会产生大量的相关向量。为了避免这种情况发生，我们对 $\boldsymbol{w}$ 中的每个元素加上先验，令 w_i 服从以 0 为均值，α_i^{-1} 为方差的正态分布，即

$$w_i \sim N(0, \frac{1}{\alpha_i})$$

故 $\boldsymbol{w}$ 的先验概率为

$$\begin{aligned} p(\boldsymbol{w}|\boldsymbol{\alpha}) &= \prod_{i=0}^{N} f(w_i; 0, \frac{1}{\alpha_i}) \\ &= \prod_{i=0}^{N} \frac{\alpha_i}{\sqrt{2\pi}} \exp(-\frac{(\alpha_i w_i)^2}{2}) \end{aligned} \tag{118}$$

其中 $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, ..., \alpha_N)^T$。根据贝叶斯定理，我们可以导出给定数据集 $\boldsymbol{t}$ 下，待优化参数 $\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2$ 的后验概率

$$\begin{aligned} p(\boldsymbol{w},\boldsymbol{\alpha},\sigma^2|\boldsymbol{t}) &= \frac{p(\boldsymbol{w},\boldsymbol{\alpha},\sigma^2,\boldsymbol{t})}{p(\boldsymbol{t})} \\ &= \frac{p(\boldsymbol{w},\boldsymbol{\alpha},\sigma^2,\boldsymbol{t})}{p(\boldsymbol{t},\boldsymbol{\alpha},\sigma^2)} \frac{p(\boldsymbol{t},\boldsymbol{\alpha},\sigma^2)}{p(\boldsymbol{t})} \\ &= p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha},\sigma^2) p(\boldsymbol{\alpha},\sigma^2|\boldsymbol{t}). \end{aligned} \tag{119}$$

因此，给定一条新的测试样本 $\boldsymbol{x}_* \in \boldsymbol{R}^m$，RVM 对其标签 t_* 预测的概率分布为

$$\begin{aligned} p(t_*|\boldsymbol{t}) &= \int p(t_*|\boldsymbol{w},\boldsymbol{\alpha},\sigma^2) p(\boldsymbol{w},\boldsymbol{\alpha},\sigma^2|\boldsymbol{t}) d\boldsymbol{w} d\boldsymbol{\alpha} d\sigma^2 \\ &= \int p(t_*|\boldsymbol{w},\boldsymbol{\alpha},\sigma^2) p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha},\sigma^2) p(\boldsymbol{\alpha},\sigma^2|\boldsymbol{t}) d\boldsymbol{w} d\boldsymbol{\alpha} d\sigma^2. \end{aligned} \tag{120}$$

到此为止，我们已经导出 RVM 用于训练及预测所需的所有参数，图5 以概率图的形式展示了参数之间的依赖关系。从图中可以看到，$\boldsymbol{w}$ 的生成依赖于 $\boldsymbol{\alpha}$，t_* 的生成依赖于 $\boldsymbol{w}, \sigma^2$。因此，RVM 的训练目标是找到一组参数 $(\boldsymbol{\alpha}_{MP}, \sigma^2_{MP})$，使得后验概率 $p(\boldsymbol{\alpha}, \sigma^2|t)$ 最大化，即

$$(\boldsymbol{\alpha}_{MP}, \sigma^2_{MP}) = \arg\max_{\boldsymbol{\alpha},\sigma^2} p(\boldsymbol{\alpha}, \sigma^2|t). \tag{121}$$

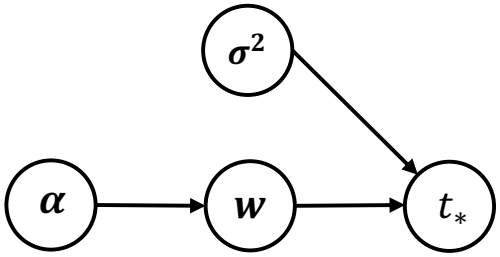


图 5: RVM 参数关系概率图

2.2 RVM 的参数估计

接下来对 Eq.(121) 进行求解。首先，根据贝叶斯定理，我们有

$$p(\boldsymbol{\alpha}, \sigma^2|\boldsymbol{t}) = \frac{p(\boldsymbol{\alpha}, \sigma^2, \boldsymbol{t})}{p(\boldsymbol{t})} = \frac{p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha})p(\sigma^2)}{p(\boldsymbol{t})} \tag{122}$$

同时，假设参数 $\boldsymbol{\alpha}, \sigma^2$ 的先验 $p(\boldsymbol{\alpha}), p(\sigma^2)$ 均服从均匀分布，则我们只需要最大化 $p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2)$，故原问题可转化为

$$(\boldsymbol{\alpha}_{MP}, \sigma^2_{MP}) = \arg\max_{\boldsymbol{\alpha}, \sigma^2} p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2). \tag{123}$$

下一步，对 $p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2)$ 进行求解，

$$\begin{aligned} p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2) &= \int p(\boldsymbol{t}, \boldsymbol{w}|\boldsymbol{\alpha}, \sigma^2)\mathrm{d}\boldsymbol{w} \\ &= \int p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{w}|\boldsymbol{\alpha})\mathrm{d}w \\ &= \int p(\boldsymbol{t}|\boldsymbol{w}, \sigma^2)p(\boldsymbol{w}|\boldsymbol{\alpha})d\boldsymbol{w}. \end{aligned} \tag{124}$$

结合 Eq.(117) 与 Eq.(118) ，进行上述积分，我们可以得到 $p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2)$ 的表达式，即

$$p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2) = (2\pi)^{-\frac{N}{2}}|\Omega|^{-\frac{1}{2}}\exp(-\frac{\boldsymbol{t}^T\Omega^{-1}t}{2}) \tag{125}$$

其中，

$$\begin{aligned} \Omega &= \sigma^2 I + \Phi A^{-1}\Phi^T, \\ A &= \mathrm{diag}(\alpha_0, \alpha_1, ..., \alpha_N). \end{aligned} \tag{126}$$

基于上式，我们还可求出 $\boldsymbol{w}$ 在给定 $\boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2$ 下的后验概率，

$$\begin{aligned} p(\boldsymbol{w}|t, \boldsymbol{\alpha}, \sigma^2) &= \frac{p(\boldsymbol{w}, \boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2)}{p(\boldsymbol{t}, \boldsymbol{\alpha}, \sigma^2)} \\ &= \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2)}{p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha}, \sigma^2)} \\ &= \frac{p(\boldsymbol{t}|\boldsymbol{w}, \boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{w}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha}, \sigma^2)}{p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2)p(\boldsymbol{\alpha}, \sigma^2)} \\ &= \frac{p(\boldsymbol{t}|\boldsymbol{w}, \sigma^2)p(\boldsymbol{w}|\boldsymbol{\alpha})}{p(\boldsymbol{t}|\boldsymbol{\alpha}, \sigma^2)}. \end{aligned} \tag{127}$$

将 Eq.(117)，Eq.(118) 与 Eq.(125)代入上式，可得到

$$p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha},\sigma^2)=(2\pi)^{-\frac{N+1}{2}}|\Sigma|^{-\frac{1}{2}}\exp(-\frac{(\boldsymbol{w}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{w}-\boldsymbol{\mu})}{2}) \tag{128}$$

其中

$$\begin{aligned}A&=\mathrm{diag}(\alpha_0,\alpha_1,...,\alpha_N),\\ \Sigma&=(\sigma^{-2}\Phi^T\Phi+A)^{-1},\\ \boldsymbol{\mu}&=\sigma^{-2}\Sigma\Phi^T\boldsymbol{t}.\end{aligned} \tag{129}$$

回到原问题。我们需要找到一组参数 $(\boldsymbol{\alpha}_{MP},\boldsymbol{\sigma}^2_{MP})$，使得转化后的目标函数 Eq.(125) 最大化。但该问题不存在闭式解，故需使用数值方法求近似解。首先是参数 $\boldsymbol{\alpha}$, 令

$$\frac{\partial p(\boldsymbol{t}|\boldsymbol{\alpha},\sigma^2)}{\partial\boldsymbol{\alpha}}=0, \tag{130}$$

可得到

$$\begin{aligned}\alpha_i^{new}&=\frac{\gamma_i}{\mu_i^2},\\ \gamma_i&=1-\alpha_i\Sigma_{ii}.\end{aligned} \tag{131}$$

其中，μ_i 为 Eq.(129) 中向量 $\boldsymbol{\mu}$ 的第 i 个元素，代表了 Eq.(128) 中 w_i 的后验均值；Σ_{ii} 是 Σ 矩阵上第 i 位对角线元素。

对于噪声方差 σ^2，同样使用 Eq.(125) 对 σ^2 进行求导并令其为 0，可以得到 σ^2 的更新值，即

$$(\sigma^2)^{new}=\frac{\|\boldsymbol{t}-\Phi\boldsymbol{\mu}\|^2}{N-\sum_{i=0}^{N}\gamma_i}, \tag{132}$$

在优化时，首先会给出 $\boldsymbol{\alpha}$ 与 σ^2 的初始值，之后通过 Eq.(135) 与 Eq.(132) 迭代更新 $\boldsymbol{\alpha}$ 和 σ^2，来达到逼近 $\boldsymbol{\alpha}_{MP}$ 和 σ^2_{MP} 的目的。大量迭代后，大部分 α_i 会增加到极大值（或称为达到了机器精确度下的无穷大），这意味着对应 w_i 的方差接近于 0，即有很强的后验确定性认为对应的 $w_i=0$；而其他 α_i 会稳定在某个有限值附近，对应的训练样本 $\boldsymbol{x}_i$ 即为**相关向量**。在预测时，我们只需要保留相关向量的数据即可完成预测过程，这体现了 RVM 的稀疏性。

2.3 RVM 的预测

完成参数估计后，可使用 RVM 对新样本进行预测。给定一条新的测试样本 $\boldsymbol{x}_*\in\boldsymbol{R}^m$，RVM 是通过输出其标签 t_* 的概率分布来实现预测，其表达式如 Eq.(120) 所示。由于 $p(\boldsymbol{\alpha},\sigma^2|\boldsymbol{t})$ 的积分是难以求解的，在实现中，我们使用 Delta Function 来近似其在 $(\boldsymbol{\alpha}_{MP},\sigma^2_{MP})$ 处的值，故 Eq.(120) 可化为

$$p(t_*|\boldsymbol{t},\boldsymbol{\alpha}_{MP},\sigma^2_{MP})=\int p(t_*|\boldsymbol{w},\sigma^2_{MP})p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha}_{MP},\sigma^2_{MP})d\boldsymbol{w}. \tag{133}$$

该积分的结果为

$$p(t_*|\boldsymbol{t},\boldsymbol{\alpha}_{MP},\sigma^2_{MP})=\frac{1}{\sqrt{2\pi}\sigma_*}\exp(-\frac{(t_*-y_*)^2}{2\sigma_*^2}) \tag{134}$$

其中

$$\begin{aligned}y_*&=\boldsymbol{\mu}^T\phi(\boldsymbol{x}_*),\\ \sigma_*^2&=\sigma^2_{MP}+\phi(\boldsymbol{x}_*)^T\Sigma\phi(\boldsymbol{x}_*),\\ \phi(\boldsymbol{x}_*)&=(1,K(\boldsymbol{x}_1,\boldsymbol{x}_*),K(\boldsymbol{x}_2,\boldsymbol{x}_*),...,K(\boldsymbol{x}_N,\boldsymbol{x}_*))^T.\end{aligned} \tag{135}$$

由此看来，RVM 的预测结果 t_* 服从以 $y(\boldsymbol{x}_*;\boldsymbol{\mu})$ 为均值，σ_*^2 为方差的正态分布，其中 $\boldsymbol{\mu}$ 为参数 $\boldsymbol{w}$ 的后验均值向量，且大部分 $\mu_i = 0$。而在实际应用中，若只需要单点预测结果，而不关心具体的概率分布信息，则可直接使用 t_* 的均值作为样本 $\boldsymbol{x}_*$ 的预测值 $\hat{t}_*$，即

$$\hat{t}_* = y(\boldsymbol{x}_*;\boldsymbol{\mu}) = \sum_{i=1}^{N} \mu_i K(\boldsymbol{x}_i, \boldsymbol{x}_*) + \mu_0. \tag{136}$$

2.4 RVM 的分类问题

RVM 分类模型的基础思想与回归模型基本相同，即使用概率建模数据标签与预测值的误差，但主要区别在于选取了适用于离散变量的概率函数。这里我们主要讨论二分类的情况，即训练集为 $T = \{(\boldsymbol{x}_i, t_i)\}_{i=1}^N$，数据特征为 $\boldsymbol{x}_i \in \boldsymbol{X} \subset \boldsymbol{R}^m$，且数据标签 $t_i \in \{0,1\}, i = 1,2,...,N$。

首先，分类 RVM 模型仍使用 Eq.(116) 的线性模型作为 t 的预测值参考，但由于分类数据的标签是离散值，因此需要对 $y(\boldsymbol{x};\boldsymbol{w})$ 进行调整。具体来说，这里使用分类问题中常用的 Sigmoid 函数对 $y(\boldsymbol{x};\boldsymbol{w})$ 进行映射，Sigmoid 函数定义为

$$\sigma(y) = \frac{1}{1+e^{-y}}.$$

该函数为定义在 $\boldsymbol{R}$ 上的单调递增函数，能将输入变量 y 映射至区间 $(0,1)$ 中，且当 $y \to -\infty$ 时，$\sigma(y) \to 0$；当 $y \to +\infty$ 时，$\sigma(y) \to 1$。

分类 RVM 与回归 RVM 的另一个区别在于概率函数的选择不同。具体来说，对于任一样本 $(\boldsymbol{x}_i, t_i)$，均假设

$$t_i \sim B(\sigma(y(\boldsymbol{x}_i;\boldsymbol{w})))$$

即 t_i 服从 $p = \sigma(y(\boldsymbol{x}_i;\boldsymbol{w}))$ 的伯努利分布。因此 t_i 的概率密度函数为

$$f(t_i;\sigma(y(\boldsymbol{x}_i;\boldsymbol{w}))) = \sigma(y(\boldsymbol{x}_i;\boldsymbol{w}))^{t_i}\left[1-\sigma(y(\boldsymbol{x}_i;\boldsymbol{w}))\right]^{1-t_i}.$$

假设所有样本是互相独立的，则似然函数可写为

$$p(\boldsymbol{t}|\boldsymbol{w}) = \prod_{i=1}^{N} \sigma(y(\boldsymbol{x}_i;\boldsymbol{w}))^{t_i}\left[1-\sigma(y(\boldsymbol{x}_i;\boldsymbol{w}))\right]^{1-t_i}. \tag{137}$$

其中，$t_i \in \{0,1\}, i = 1,2,...,N$。与回归问题不同，这里不需要使用额外的“噪音”方差 σ^2。

分类 RVM 的求解思想与上述回归问题基本相同，但 Eq.(137) 难以直接求出，且权重 $\boldsymbol{w}$ 的后验概率 $p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha}$ 以及其对应的 $p(\boldsymbol{t}|\boldsymbol{\alpha})$ 同样不存在闭式解。在这里，我们使用的是一种基于拉普拉斯方法的近似过程实现参数的估计，该方法的具体流程如下。

1. 固定当前 $\boldsymbol{\alpha}$，求后验概率 $p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha})$ 最大处对应的 $\boldsymbol{w}$，即

$$\boldsymbol{w}_{MP} = \arg\max_{\boldsymbol{w}} p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha}). \tag{138}$$

由于 $p(\boldsymbol{w}|\boldsymbol{t},\boldsymbol{\alpha}) \propto p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})$，结合 Eq.(137) 与 Eq.(118)，故 Eq.(138)中最大化的目标函数可等价于

$$\log\left[p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})\right] = \sum_{i=1}^{N}\left[t_i \log y_i + (1 - t_i \log(1-y_i))\right] - \frac{1}{2}\boldsymbol{w}^T\boldsymbol{A}\boldsymbol{w}, \tag{139}$$

其中 $y_i = \sigma(y(\boldsymbol{x}_i;\boldsymbol{w}))$。观察 Eq.(139) 可以发现，该函数实际上是带正则项约束的交叉熵损失，这也是二分类问题中常用的损失函数，但二分类问题中常用的交叉熵损失往往是在 Eq.(139) 上取负

值，并使用反向传播等算法不断优化使其达到最小值。而在分类 RVM 中，使用的是二阶牛顿法来求解 $\boldsymbol{w}_{MP}$，方法如下

$$\begin{aligned}\boldsymbol{H} &= \nabla_{\boldsymbol{w}}\nabla_{\boldsymbol{w}}\log\left[p(\boldsymbol{t}|\boldsymbol{w})p(\boldsymbol{w}|\boldsymbol{\alpha})\right] = (-\Phi^T\boldsymbol{B}\Phi - \boldsymbol{A}\boldsymbol{w})^{-1},\\ \triangle\boldsymbol{w} &= -\boldsymbol{H}^{-1}\boldsymbol{g},\\ \boldsymbol{w}_{MP}^{new} &= \boldsymbol{w}_{MP} + \triangle\boldsymbol{w},\end{aligned}$$

其中

$$\begin{aligned}\boldsymbol{B} &= diag(\beta_1, \beta_2, ..., \beta_N),\\ \beta_i &= \sigma(y(\boldsymbol{x}_i))\left[1 - \sigma(y(\boldsymbol{x}_i))\right].\end{aligned}$$

2. 拉普拉斯近似使用高斯分布近似 $p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha})$，该高斯分布的均值为 $\boldsymbol{w}_{MP}$，协方差矩阵 Σ 则使用第 1 步中的黑塞矩阵计算得到，即

$$\Sigma = (-\boldsymbol{H}|_{\boldsymbol{w_{MP}}})^{-1} = (\Phi^T\boldsymbol{B}\Phi + \boldsymbol{A})^{-1} \tag{140}$$

3. 使用步骤 2 中得到的高斯分布更新参数 $\boldsymbol{\alpha}$，方法与 Eq.(135)相同。

References

[1] Colin Campbell. "Kernel methods: a survey of current techniques". In: *Neurocomputing* 48.1-4 (2002), pp. 63–84 (cit. on p. 13).

[2] Corinna Cortes and Vladimir Vapnik. "Support-vector networks". In: *Machine learning* 20.3 (1995), pp. 273–297 (cit. on p. 7).

[3] Harris Drucker et al. "Support vector regression machines". In: *Advances in neural information processing systems*. 1997, pp. 155–161 (cit. on p. 20).

[4] Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. "A practical guide to support vector classification". In: (2003) (cit. on p. 15).

[5] S Sathiya Keerthi and Chih-Jen Lin. "Asymptotic behaviors of support vector machines with Gaussian kernel". In: *Neural computation* 15.7 (2003), pp. 1667–1689 (cit. on p. 14).

[6] John Platt. "Sequential minimal optimization: A fast algorithm for training support vector machines". In: (1998) (cit. on pp. 15, 19).

[7] Michael E Tipping. "Sparse Bayesian learning and the relevance vector machine". In: *Journal of machine learning research* 1.Jun (2001), pp. 211–244 (cit. on p. 23).

[8] Yan-fei Zhu et al. "Mixtures of kernels for SVM modeling". In: *International Conference on Natural Computation*. Springer. 2005, pp. 601–607 (cit. on p. 15).

[9] 乃扬 and 英杰. *数据挖掘中的新方法: 支持向量机*. 科学出版社, 2004 (cit. on p. 13).

[10] 李航 et al. *统计学习方法*. 北京: 清华大学出版社, 2012 (cit. on p. 1).