支持向量机宝宝巴士

Apple Zhang

Shenzhen University

April 14, 2025

目录

目录

符号定义

- 第i个训练样本: $\boldsymbol{x}_i = [x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(d)}]^\top \in \mathbb{R}^d$.
- 第i个训练样本的标签: $y_i \in \{-1, 1\}$.
- 训练样本矩阵: $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times n}$.
- 训练样本标签向量: $\boldsymbol{y} = [y_1, y_2, \cdots, y_n] \in \{-1, 1\}^n$.
- L_2范数:

$$\|\boldsymbol{x}\|_2^2 = \sum_{i=1}^{d} (x^{(i)})^2 = \boldsymbol{x}^\top \boldsymbol{x}.$$

- 函数 $f(\cdot)$ 在 $\boldsymbol{x}$ 点处的梯度

$$\nabla f(\boldsymbol{x}) = \left[\frac{\partial f}{\partial x^{(1)}}, \frac{\partial f}{\partial x^{(2)}}, \cdots, \frac{\partial f}{\partial x^{(d)}}\right]^\top.$$

例如：$f(\boldsymbol{x}) = \boldsymbol{x}^\top \boldsymbol{x}$，则

$$\frac{\partial f}{\partial x^{(i)}} = \frac{\partial \left((x^{(i)})^2 + \sum_{j\neq i}(x^{(j)})^2\right)}{\partial x^{(i)}} = 2x^{(i)},$$

所以 $\nabla f(\boldsymbol{x}) = 2\boldsymbol{x}$.

什么是超平面?

- 在$\mathbb{R}^2$空间中, 一条直线可表示为集合

$$\left\{(x^{(1)}, x^{(2)}) \mid w_1 x^{(1)} + w_2 x^{(2)} + b = 0\right\}$$

- 在$\mathbb{R}^3$空间中, 一个平面可表示为集合

$$\left\{(x^{(1)}, x^{(2)}, x^{(3)}) \mid w_1 x^{(1)} + w_2 x^{(2)} + w_3 x^{(3)} + b = 0\right\}.$$

- 在$\mathbb{R}^d$空间中, 一个超平面可表示为集合

$$\{\boldsymbol{x} \mid \boldsymbol{w}^\top \boldsymbol{x} + b = 0\}$$

即:

$$\left\{(x^{(1)}, x^{(2)}, \cdots, x^{(d)}) \mid w_1 x^{(1)} + w_2 x^{(2)} + \cdots + w_d x^{(d)} + b = 0\right\}$$

我们分别称 $\boldsymbol{w} \in \mathbb{R}^d$ 和 $b \in \mathbb{R}$ 为超平面的法向量与偏移.

数据点到超平面的距离

- 在 $\mathbb{R}^2$ 空间中, 数据点 $\boldsymbol{x}_i = [x_i^{(1)}, x_i^{(2)}]^\top$ 到一条直线的距离:

$$\frac{|w_1 x_i^{(1)} + w_2 x_i^{(2)} + b|}{\sqrt{w_1^2 + w_2^2}}.$$

- 在 $\mathbb{R}^3$ 空间中, 数据点 $\boldsymbol{x}_i = [x_i^{(1)}, x_i^{(2)}, x_i^{(3)}]^\top$ 到一个平面的距离:

$$\frac{|w_1 x_i^{(1)} + w_2 x_i^{(3)} + w_2 x_i^{(3)} + b|}{\sqrt{w_1^2 + w_2^2 + w_3^2}}.$$

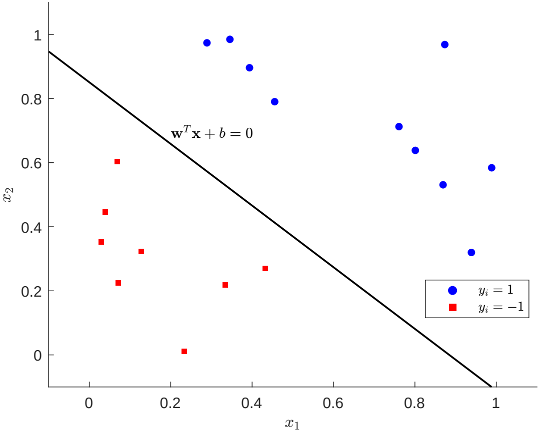
- 在 $\mathbb{R}^d$ 空间中, 数据点 $\boldsymbol{x}_i = [x_i^{(1)}, x_i^{(2)}, \cdots, x_i^{(d)}]^\top$ 到超平面的距离:

$$\frac{|w_1 x_i^{(1)} + w_2 x_i^{(2)} + \cdots + w_d x_i^{(d)} + b|}{\sqrt{w_1^2 + w_2^2 + \cdots + w_d^2}}$$
$$= \frac{|\boldsymbol{w}^\top \boldsymbol{x}_i + b|}{\|\boldsymbol{w}\|_2}.$$

目录

从最简单的线性可分出发吧

问题:
对于一个二分类问题，假设数据是完全线性可分的，如何找到一个超平面把两类数据分开?

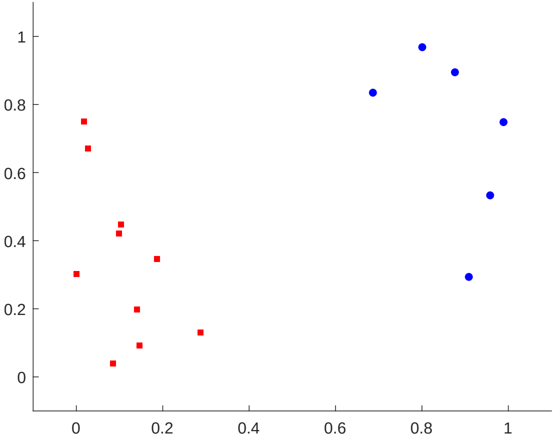


关键概念:

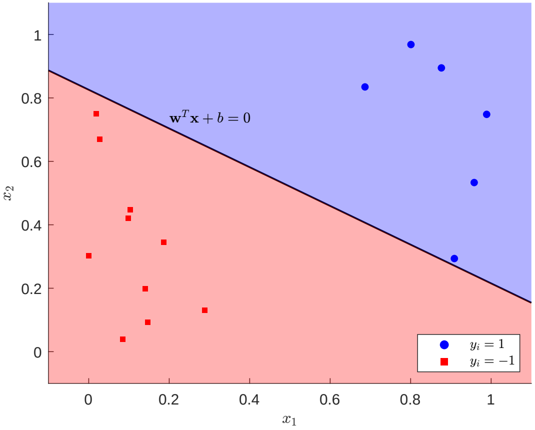
- 超平面: $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$.
- 预测函数:
  $h(\boldsymbol{x}) = \mathrm{sign}(\boldsymbol{w}^\top \boldsymbol{x} + b)$.
- 完全线性可分: 存在至少一个超平面，使得
  $\forall i,\ y_i = h(\boldsymbol{x}_i)$.

深圳大学 SHENZHEN UNIVERSITY

如何对这个数据分类

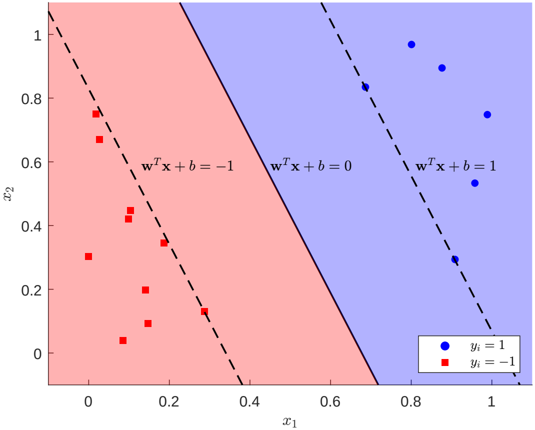


如何对这个数据分类



最大间隔分类器: 支持向量机

SVM 寻找最大化两个类间的间隔的超平面[CV95].

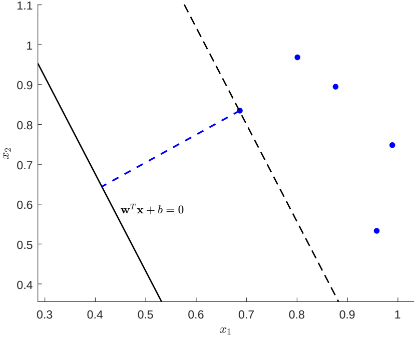


SVM 的优化目标

定义 (间隔)

样本集合 $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n\}$ 到超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ 的间隔，等于离超平面最近的那个点到超平面的距离:

$$\rho = \min_{i=1,2,\cdots,n} \frac{|\boldsymbol{w}^\top \boldsymbol{x}_i + b|}{\|\boldsymbol{w}\|_2}. \tag{1}$$

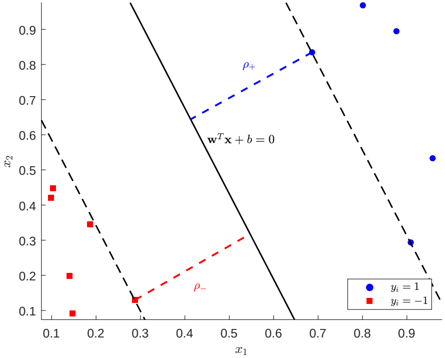


SHENZHEN UNIVERSITY

SVM 的优化目标

计算正类样本和负类样本到超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ 的间隔之和:

$$\rho = \rho_+ + \rho_-. \tag{2}$$



一般来说，我们希望正负类到超平面的间隔相等，即 $\rho_+ = \rho_-$，此时分类超平面位于两类样本 “空白区域” 的正中间.

SVM 的优化目标

根据下面的步骤计算出总间隔吧:

1. 不失一般性，设靠近正类的虚线为 $\boldsymbol{w}^\top \boldsymbol{x} + b = 1$;（啊？这是为何？）
2. 此时可以计算正类样本到超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ 的间隔:

$$\begin{aligned}
\rho_+ &= \min_{i:y_i=+1} \frac{|\boldsymbol{w}^\top \boldsymbol{x}_i + b|}{\|\boldsymbol{w}\|_2} \\
&= \min_{i:y_i=+1} \frac{\boldsymbol{w}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{w}\|_2}, \qquad \text{因为 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 1 > 0, \forall i : y_i = +1, \\
&= \frac{1}{\|\boldsymbol{w}\|_2}, \qquad \text{因为存在某个 } i_0 \text{ 满足 } \boldsymbol{w}^\top \boldsymbol{x}_{i_0} + b = 1.
\end{aligned}$$

3. 由于我们设置了 $\rho_+ = \rho_-$，所以:

$$\rho = \rho_+ + \rho_- = 2\rho_+ = \frac{2}{\|\boldsymbol{w}\|_2}.$$

深圳大学 SHENZHEN UNIVERSITY

SVM 的优化目标

根据下面的步骤计算出总间隔吧:

1. 不失一般性，设靠近正类的虚线为 $\boldsymbol{w}^\top \boldsymbol{x} + b = 1$; （啊？这是为何？）
2. 此时可以计算正类样本到超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ 的间隔:

$$\begin{aligned}\rho_+ &= \min_{i:y_i=+1} \frac{|\boldsymbol{w}^\top \boldsymbol{x}_i + b|}{\|\boldsymbol{w}\|_2} \\ &= \min_{i:y_i=+1} \frac{\boldsymbol{w}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{w}\|_2}, \qquad \text{因为 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 1 > 0, \forall i : y_i = +1, \\ &= \frac{1}{\|\boldsymbol{w}\|_2}, \qquad \text{因为存在某个 } i_0 \text{ 满足 } \boldsymbol{w}^\top \boldsymbol{x}_{i_0} + b = 1.\end{aligned}$$

3. 由于我们设置了 $\rho_+ = \rho_-$，所以:

$$\rho = \rho_+ + \rho_- = 2\rho_+ = \frac{2}{\|\boldsymbol{w}\|_2}.$$

深圳大学 SHENZHEN UNIVERSITY

SVM 的优化目标

根据下面的步骤计算出总间隔吧:

1. 不失一般性，设靠近正类的虚线为 $\boldsymbol{w}^\top \boldsymbol{x} + b = 1$; （啊？这是为何？）
2. 此时可以计算正类样本到超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ 的间隔:

$$\begin{aligned}\rho_+ &= \min_{i:y_i=+1} \frac{|\boldsymbol{w}^\top \boldsymbol{x}_i + b|}{\|\boldsymbol{w}\|_2} \\ &= \min_{i:y_i=+1} \frac{\boldsymbol{w}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{w}\|_2}, && \text{因为 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 1 > 0, \forall i: y_i = +1, \\ &= \frac{1}{\|\boldsymbol{w}\|_2}, && \text{因为存在某个 } i_0 \text{ 满足 } \boldsymbol{w}^\top \boldsymbol{x}_{i_0} + b = 1.\end{aligned}$$

3. 由于我们设置了 $\rho_+ = \rho_-$，所以:

$$\rho = \rho_+ + \rho_- = 2\rho_+ = \frac{2}{\|\boldsymbol{w}\|_2}.$$

深圳大学 SHENZHEN UNIVERSITY

SVM 的优化目标

根据下面的步骤计算出总间隔吧:

1. 不失一般性，设靠近正类的虚线为 $\boldsymbol{w}^\top \boldsymbol{x} + b = 1$; （啊？这是为何？）
2. 此时可以计算正类样本到超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$ 的间隔:

$$
\begin{aligned}
\rho_+ &= \min_{i:y_i=+1} \frac{|\boldsymbol{w}^\top \boldsymbol{x}_i + b|}{\|\boldsymbol{w}\|_2} \\
&= \min_{i:y_i=+1} \frac{\boldsymbol{w}^\top \boldsymbol{x}_i + b}{\|\boldsymbol{w}\|_2}, \qquad \text{因为 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 1 > 0, \forall i : y_i = +1, \\
&= \frac{1}{\|\boldsymbol{w}\|_2}, \qquad \text{因为存在某个 } i_0 \text{ 满足 } \boldsymbol{w}^\top \boldsymbol{x}_{i_0} + b = 1.
\end{aligned}
$$

3. 由于我们设置了 $\rho_+ = \rho_-$，所以:

$$
\rho = \rho_+ + \rho_- = 2\rho_+ = \frac{2}{\|\boldsymbol{w}\|_2}.
$$

SVM 的优化问题

SVM 的目标就是找到 $\boldsymbol{w}$ 和 b，确定超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$，使得间隔 ρ 最大化：

$$\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|_2}, \quad \text{s.t. } |\boldsymbol{w}^\top \boldsymbol{x}_i + b| \geq 1. \tag{3}$$

我们还可以进一步化简，根据以下事实：

1. 一个函数 $f(x) > 0$，那么最大化 $f(x)$ 等价于最小化它的倒数；

$$\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|_2} \Leftrightarrow \min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2.$$

2. $|\boldsymbol{w}^\top \boldsymbol{x}_i + b| = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$，因为

$$|\boldsymbol{w}^\top \boldsymbol{x}_i + b| = \begin{cases} \boldsymbol{w}^\top \boldsymbol{x}_i + b = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), & \text{如果 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 1, \\ -(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), & \text{如果 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \leq -1, \end{cases}$$

最终将得到 “硬间隔” SVM 的优化问题:

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1.$$

SVM 的优化问题

SVM 的目标就是找到 $\boldsymbol{w}$ 和 b，确定超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$，使得间隔 ρ 最大化：

$$\max_{\boldsymbol{w}, b} \frac{2}{\|\boldsymbol{w}\|_2}, \quad \text{s.t.} \ |\boldsymbol{w}^\top \boldsymbol{x}_i + b| \geq 1. \tag{3}$$

我们还可以进一步化简，根据以下事实：

1. 一个函数 $f(x) > 0$，那么最大化 $f(x)$ 等价于最小化它的倒数；

$$\max_{\boldsymbol{w}, b} \frac{2}{\|\boldsymbol{w}\|_2} \Leftrightarrow \min_{\boldsymbol{w}, b} \frac{1}{2} \|\boldsymbol{w}\|_2^2.$$

2. $|\boldsymbol{w}^\top \boldsymbol{x}_i + b| = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$，因为

$$|\boldsymbol{w}^\top \boldsymbol{x}_i + b| = \begin{cases} \boldsymbol{w}^\top \boldsymbol{x}_i + b = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), & \text{如果 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 1, \\ -(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), & \text{如果 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \leq -1, \end{cases}$$

最终将得到 “硬间隔” SVM 的优化问题:

$$\min_{\boldsymbol{w}, b} \frac{1}{2} \|\boldsymbol{w}\|_2^2, \quad \text{s.t.} \ y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1.$$

SVM 的优化问题

SVM 的目标就是找到 $\boldsymbol{w}$ 和 b，确定超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$，使得间隔 ρ 最大化：

$$\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|_2}, \quad \text{s.t. } |\boldsymbol{w}^\top \boldsymbol{x}_i + b| \geq 1. \tag{3}$$

我们还可以进一步化简，根据以下事实：

1. 一个函数 $f(x) > 0$，那么最大化 $f(x)$ 等价于最小化它的倒数；

$$\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|_2} \Leftrightarrow \min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2.$$

2. $|\boldsymbol{w}^\top \boldsymbol{x}_i + b| = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$，因为

$$|\boldsymbol{w}^\top \boldsymbol{x}_i + b| = \begin{cases} \boldsymbol{w}^\top \boldsymbol{x}_i + b = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), & \text{如果 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 1, \\ -(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), & \text{如果 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \leq -1, \end{cases}$$

最终将得到 “硬间隔” SVM 的优化问题:

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1.$$

深圳大学 SHENZHEN UNIVERSITY

SVM 的优化问题

SVM 的目标就是找到 $\boldsymbol{w}$ 和 b，确定超平面 $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$，使得间隔 ρ 最大化：

$$\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|_2}, \quad \text{s.t. } |\boldsymbol{w}^\top \boldsymbol{x}_i + b| \geq 1. \tag{3}$$

我们还可以进一步化简，根据以下事实：

1. 一个函数 $f(\boldsymbol{x}) > 0$，那么最大化 $f(\boldsymbol{x})$ 等价于最小化它的倒数；

$$\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|_2} \Leftrightarrow \min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2.$$

2. $|\boldsymbol{w}^\top \boldsymbol{x}_i + b| = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$，因为

$$|\boldsymbol{w}^\top \boldsymbol{x}_i + b| = \begin{cases} \boldsymbol{w}^\top \boldsymbol{x}_i + b = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), & \text{如果 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \geq 1, \\ -(\boldsymbol{w}^\top \boldsymbol{x}_i + b) = y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b), & \text{如果 } \boldsymbol{w}^\top \boldsymbol{x}_i + b \leq -1, \end{cases}$$

最终将得到 “硬间隔” SVM 的优化问题:

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1.$$

深圳大学 SHENZHEN UNIVERSITY

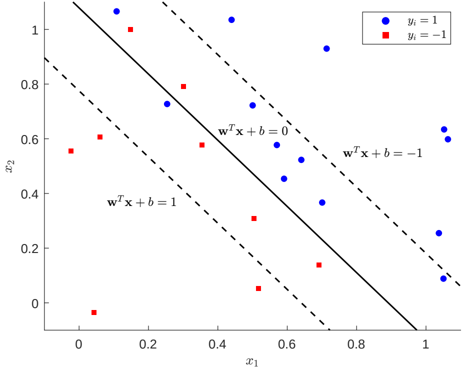
目录

对…对吗？

这是我们刚刚得到的 “硬间隔” SVM:

$$\min_{\boldsymbol{w},b} \frac{1}{2}\|\boldsymbol{w}\|_2^2, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1.$$

它的要求是数据完全线性可分，但一般这个对实际数据基本不成立，所以这个问题在的实际问题中是无解的[1]. 我们更多使用的是**软间隔SVM**:



[1]这是一个思想钢印，如果你是理论方面的专家，你可能已经知道 over-parameterized 的非线性 SVM 有机会实现完美的 “训练样本插值”，此时的模型就是硬间隔 SVM.

哦对的对的

软间隔 SVM 允许一部分样本不满足 $y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1$，它的优化问题是:

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + c\sum_{i=1}^{n}\xi_i, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0.$$

其中 $c > 0$ 是一个超参数，ξ_i 是松弛变量（它是一个优化变量），表示第 i 个样本违反了约束条件的程度，可以把它视作一种惩罚，我们希望惩罚越小越好.

软间隔 SVM 才是我们通常意义上提及的 SVM.

几何意义

如果我们单独看关于每一个 ξ_i 的最小化问题，也就是

$$\min_{\xi_i} \xi_i, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0.$$

可以得到两个结论:

- 如果 $y_i(w^\top x_i + b) \geq 1$，那么 $\xi_i = 0$（啊？这是为何？）；也就是说，“如果满足条件 $y_i(w^\top x_i + b) \geq 1$ ，那么就没有惩罚”.
- 如果 $y_i(w^\top x_i + b) < 1$，那么 $\xi_i = 1 - y_i(w^\top x_i + b)$（啊？这是为何？）；说明此时“产生了 $1 - y_i(w^\top x_i + b)$ 的惩罚”.

几何意义

如果我们单独看关于每一个 ξ_i 的最小化问题，也就是

$$\min_{\xi_i} \xi_i, \quad \text{s.t.} \; y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i, \; \xi_i \geq 0.$$

可以得到两个结论:

- 如果 $y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1$，那么 $\xi_i = 0$ （啊？这是为何？）；也就是说，**“如果满足条件 $y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1$ ，那么就没有惩罚”.**
- 如果 $y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) < 1$，那么 $\xi_i = 1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$ （啊？这是为何？）；说明此时 **“产生了 $1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$ 的惩罚”.**

深圳大学 SHENZHEN UNIVERSITY

几何意义

如果我们单独看关于每一个 ξ_i 的最小化问题，也就是

$$\min_{\xi_i} \xi_i, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0.$$

可以得到两个结论:

- 如果 $y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1$，那么 $\xi_i = 0$ （啊？这是为何？）；也就是说，**“如果满足条件 $y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1$，那么就没有惩罚”**.
- 如果 $y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) < 1$，那么 $\xi_i = 1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$ （啊？这是为何？）；说明此时 **“产生了 $1 - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)$ 的惩罚”**.

几何意义

接下来，我们要解决这个优化问题：

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + c\sum_{i=1}^{n}\xi_i, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0. \tag{4}$$

很显然，这个优化问题的解由以下定理给出：

定理

问题 (4) 的最优解 $(\boldsymbol{w}^\star, b^\star, \boldsymbol{\xi}^\star)$ 满足：

$$\boldsymbol{w}^\star = \sum_{i=1}^{n}\alpha_i^\star y_i \boldsymbol{x}_i; \quad b^\star = y_j - \boldsymbol{x}_j^\top \boldsymbol{w}^\star, \forall j : 0 < \alpha_j^\star < c; \tag{5}$$

$$\boldsymbol{\alpha}^\star = \underset{0\leq\alpha_i\leq c}{\operatorname{argmax}} \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j, \text{ s.t. } \sum_{i=1}^{n}\alpha_i y_i = 0. \tag{6}$$

接下来解释一下 “显然” 的部分.

深圳大学 SHENZHEN UNIVERSITY

几何意义

接下来，我们要解决这个优化问题：

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + c\sum_{i=1}^{n}\xi_i, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0. \tag{4}$$

很显然，这个优化问题的解由以下定理给出：

定理

问题 (4) 的最优解 $(\boldsymbol{w}^\star, b^\star, \boldsymbol{\xi}^\star)$ 满足：

$$\boldsymbol{w}^\star = \sum_{i=1}^{n}\alpha_i^\star y_i \boldsymbol{x}_i; \quad b^\star = y_j - \boldsymbol{x}_j^\top \boldsymbol{w}^\star, \forall j : 0 < \alpha_j^\star < c; \tag{5}$$

$$\boldsymbol{\alpha}^\star = \operatorname*{argmax}_{0\leq\alpha_i\leq c} \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j, \text{ s.t. } \sum_{i=1}^{n}\alpha_i y_i = 0. \tag{6}$$

接下来解释一下 “显然” 的部分.

几何意义

接下来，我们要解决这个优化问题：

$$\min_{\boldsymbol{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + c\sum_{i=1}^{n}\xi_i, \quad \text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0. \tag{4}$$

很显然，这个优化问题的解由以下定理给出：

定理

问题 (4) 的最优解 $(\boldsymbol{w}^\star, b^\star, \boldsymbol{\xi}^\star)$ 满足：

$$\boldsymbol{w}^\star = \sum_{i=1}^{n}\alpha_i^\star y_i \boldsymbol{x}_i; \quad b^\star = y_j - \boldsymbol{x}_j^\top \boldsymbol{w}^\star, \forall j : 0 < \alpha_j^\star < c; \tag{5}$$

$$\boldsymbol{\alpha}^\star = \operatorname*{argmax}_{0\leq\alpha_i\leq c} \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j, \text{ s.t. } \sum_{i=1}^{n}\alpha_i y_i = 0. \tag{6}$$

接下来解释一下 “显然” 的部分.

带不等式约束的优化问题

定义以下带不等式约束的优化问题:

$$p^{\star} = \min_{\boldsymbol{u}} f_0(\boldsymbol{u}), \quad \text{s.t. } f_i(\boldsymbol{u}) \leq 0, \ \forall i \in \{1, 2, \cdots, n\}. \tag{7}$$

定义 (7) 对应的拉格朗日函数为

$$L(\boldsymbol{u}, \boldsymbol{\alpha}) = f_0(\boldsymbol{u}) + \sum_{i=1}^{n} \alpha_i f_i(\boldsymbol{u}). \tag{8}$$

定义 (7) 的对偶问题为

$$d^{\star} = \max_{\boldsymbol{\alpha}} \min_{\boldsymbol{u}} L(\boldsymbol{u}, \boldsymbol{\alpha}), \quad \text{s.t. } \alpha_i \geq 0, \ \forall i \in \{1, 2, \cdots, n\}. \tag{9}$$

深圳大学 SHENZHEN UNIVERSITY

带不等式约束的优化问题

定义以下带不等式约束的优化问题:

$$p^\star = \min_{\boldsymbol{u}} f_0(\boldsymbol{u}), \quad \text{s.t. } f_i(\boldsymbol{u}) \leq 0, \ \forall i \in \{1, 2, \cdots, n\}. \tag{7}$$

定义 (7) 对应的拉格朗日函数为

$$L(\boldsymbol{u}, \boldsymbol{\alpha}) = f_0(\boldsymbol{u}) + \sum_{i=1}^{n} \alpha_i f_i(\boldsymbol{u}). \tag{8}$$

定义 (7) 的对偶问题为

$$d^\star = \max_{\boldsymbol{\alpha}} \min_{\boldsymbol{u}} L(\boldsymbol{u}, \boldsymbol{\alpha}), \quad \text{s.t. } \alpha_i \geq 0, \ \forall i \in \{1, 2, \cdots, n\}. \tag{9}$$

带不等式约束的优化问题

原问题：

$$p^\star = \min_{\boldsymbol{u}} f_0(\boldsymbol{u}), \quad \text{s.t. } f_i(\boldsymbol{u}) \leq 0, \ \forall i \in \{1, 2, \cdots, n\}.$$

对偶问题：

$$d^\star = \max_{\boldsymbol{\alpha}} \min_{\boldsymbol{u}} L(\boldsymbol{u}, \boldsymbol{\alpha}), \quad \text{s.t. } \alpha_i \geq 0, \ \forall i \in \{1, 2, \cdots, n\}.$$

如果一个问题满足**强对偶性**，那么 $p^\star = d^\star$，且原问题的解等于对偶问题的解.

SVM 的问题满足强对偶性，因此可以通过求解对偶问题得到原问题的解.

SVM 的对偶问题

SVM 的拉格朗日函数:

$$L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + c\sum_{i=1}^{n}\xi_i + \sum_{i=1}^{n}\alpha_i[1 - \xi_i - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)] - \sum_{i=1}^{n}\beta_i\xi_i.$$

SVM 的对偶问题

$$\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} L(\boldsymbol{w},b,\boldsymbol{\xi},\boldsymbol{\alpha},\boldsymbol{\beta}), \quad \text{s.t. } \alpha_i \geq 0,\ \beta_i \geq 0,\ \forall i \in \{1,2,\cdots,n\}.$$

- 第一步：对于 $\boldsymbol{w}, b, \boldsymbol{\xi}$ 最小化 $L(\boldsymbol{w}, b, \boldsymbol{\alpha})$:

$$\nabla_{\boldsymbol{w}} L = 0 \Rightarrow \boldsymbol{w} = \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i, \quad \nabla_b L = 0 \Rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0,$$
$$\nabla_{\boldsymbol{\xi}} L = 0 \Rightarrow c - \alpha_i - \beta_i = 0.$$

- 第二步：计算 $\min_{\boldsymbol{w},b,\boldsymbol{\xi}} L$，即把以上条件代入，则对偶问题为

$$\max_{0\leq\alpha_i\leq c} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i\alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j, \ \text{s.t. } \sum_{i=1}^{n} \alpha_i y_i = 0.$$

深圳大学 SHENZHEN UNIVERSITY

SVM 的对偶问题

写成矩阵形式:

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0. \tag{10}$$

这里 $\mathbf{1}$ 是全1的列向量, $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$的每一个元素是

$$Q_{ij} = y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{11}$$

- 对偶问题 (12) 的解 $\alpha^\star$ 是稀疏的，意味着其中有许多零元素 $\alpha_i = 0$.
- 如果 $\alpha_i > 0$, 则其对应的训练样本 x_i 称为支持向量(**Support Vector**).
- 通过 KKT 条件可以知道对任意的 $i : 0 < \alpha_i < c$，有 $y_i(x_i^\top \boldsymbol{w}^\star + b) = 1$，由此可以得到 $b^\star = y_i^{-1} - x_i^\top \boldsymbol{w}^\star$.

SVM 的对偶问题

写成矩阵形式:

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0. \tag{10}$$

这里 $\mathbf{1}$ 是全1的列向量, $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$的每一个元素是

$$Q_{ij} = y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{11}$$

- 对偶问题 (12) 的解 $\boldsymbol{\alpha}^\star$ 是稀疏的，意味着其中有许多零元素 $\alpha_i = 0$.
- 如果 $\alpha_i > 0$, 则其对应的训练样本 x_i 称为支持向量(Support Vector).
- 通过 KKT 条件可以知道对任意的 $i : 0 < \alpha_i < c$，有 $y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^\star + b) = 1$，由此可以得到 $b^\star = y_i^{-1} - \boldsymbol{x}_i^\top \boldsymbol{w}^\star$.

写成矩阵形式:

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0. \tag{10}$$

这里 $\mathbf{1}$ 是全1的列向量, $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$的每一个元素是

$$Q_{ij} = y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{11}$$

- 对偶问题 (12) 的解 $\boldsymbol{\alpha}^\star$ 是稀疏的，意味着其中有许多零元素 $\alpha_i = 0$.
- 如果 $\alpha_i > 0$, 则其对应的训练样本 $\boldsymbol{x}_i$ 称为支持向量(**Support Vector**).
- 通过 KKT 条件可以知道对任意的 $i : 0 < \alpha_i < c$，有 $y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^\star + b) = 1$，由此可以得到 $b^\star = y_i^{-1} - \boldsymbol{x}_i^\top \boldsymbol{w}^\star$.

SVM 的对偶问题

写成矩阵形式:

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0. \tag{10}$$

这里 $\mathbf{1}$ 是全1的列向量, $\boldsymbol{Q} \in \mathbb{R}^{n\times n}$的每一个元素是

$$Q_{ij} = y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j. \tag{11}$$

- 对偶问题 (12) 的解 $\boldsymbol{\alpha}^\star$ 是稀疏的，意味着其中有许多零元素 $\alpha_i = 0$.
- 如果 $\alpha_i > 0$, 则其对应的训练样本 $\boldsymbol{x}_i$ 称为支持向量(**Support Vector**).
- 通过 KKT 条件可以知道对任意的 $i : 0 < \alpha_i < c$，有 $y_i(\boldsymbol{x}_i^\top \boldsymbol{w}^\star + b) = 1$，由此可以得到 $b^\star = y_i^{-1} - \boldsymbol{x}_i^\top \boldsymbol{w}^\star$.

如何求解对偶问题

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0.$$

- 喜欢用 MATLAB 的：`quadprog`;
- 喜欢用 Python 的：`cvxopt.solvers.qp`;
- 其实你不用自己实现：LIBSVM 库 [CL11]，由 C++ 实现； Python 中对应的 scikit-learn 接口为 `sklearn.svm.SVC`.

如何求解对偶问题

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0.$$

- 喜欢用 MATLAB 的：`quadprog`;
- 喜欢用 Python 的：`cvxopt.solvers.qp`;
- 其实你不用自己实现：LIBSVM 库 [CL11]，由 C++ 实现； Python 中对应的 scikit-learn 接口为 `sklearn.svm.SVC`.

深圳大学 SHENZHEN UNIVERSITY

如何求解对偶问题

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c, \ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0.$$

- 喜欢用 MATLAB 的：`quadprog`;
- 喜欢用 Python 的：`cvxopt.solvers.qp`;
- 其实你不用自己实现：LIBSVM 库 [CL11]，由 C++ 实现； Python 中对应的 scikit-learn 接口为 `sklearn.svm.SVC`.

如何求解对偶问题

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c, \ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0.$$

- 喜欢用 MATLAB 的：`quadprog`;
- 喜欢用 Python 的：`cvxopt.solvers.qp`;
- 其实你不用自己实现：LIBSVM 库 [CL11]，由 C++ 实现；Python 中对应的 scikit-learn 接口为 `sklearn.svm.SVC`.

预测新样本的标签

$$\boldsymbol{\alpha}^\star = \underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \ \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c, \ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0.$$

当我们获得最优解 $\boldsymbol{\alpha}^\star$ 时，我们有

$$\boldsymbol{w}^\star = \sum_{i=1}^{n} \alpha_i^\star y_i \boldsymbol{x}_i; \quad b^\star = y_j - \boldsymbol{x}_j^\top \boldsymbol{w}^\star, \ \forall j : 0 < \alpha_j^\star < c;$$

而对新样本 $\boldsymbol{x}$ 的预测为

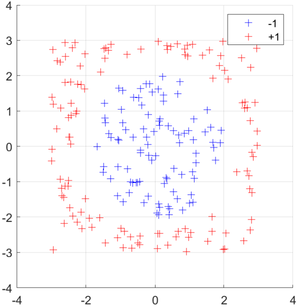
$$\begin{aligned} h(\boldsymbol{x}) &= \operatorname{sign}(\boldsymbol{x}^\top \boldsymbol{w}^\star + b^\star) \\ &= \operatorname{sign}\left(\sum_{i=1}^{n} \alpha_i^\star y_i \boldsymbol{x}_i^\top \boldsymbol{x} + b^\star\right) = \operatorname{sign}\left(\sum_{i:\alpha_i \neq 0} \alpha_i^\star y_i \boldsymbol{x}_i^\top \boldsymbol{x} + b^\star\right). \end{aligned}$$

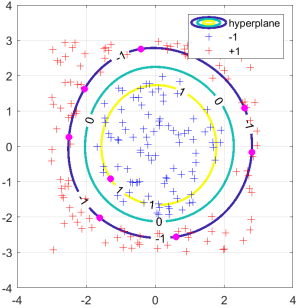
深圳大学 SHENZHEN UNIVERSITY

目录

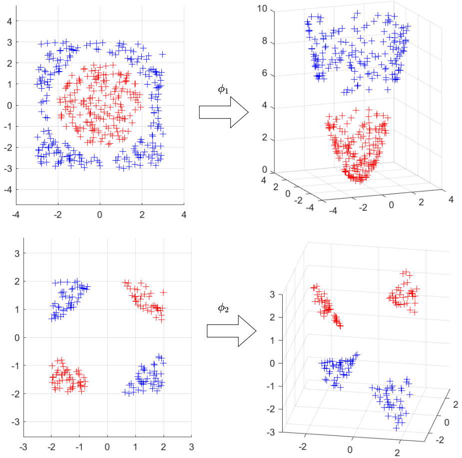
非线性分类: 核 SVM

那么如果是非线性分布的数据呢？
核 SVM: 通过核方法实现非线性分类.





非线性问题能否变为线性问题?



如何处理非线性数据

回到 SVM 的对偶问题

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0. \tag{12}$$

根据我们之前的推导，这里的 $\boldsymbol{Q}$ 矩阵为 $Q_{ij} = y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j$.

如果将其中的 x 替换为 $\phi(x)$，也就是让 $Q_{ij} = y_i y_j \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j)$，就相当于:

1. 先对所有数据施加非线性映射 $\phi(\cdot)$，
2. 再用 $\phi(\boldsymbol{x}_i)$ 作为输入来训练线性 SVM.

再进一步，定义一个函数 $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^\top \phi(\boldsymbol{x}')$，则有 $Q_{ij} = y_i y_j \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)$. 这就是核 SVM 的初步思想.

如何处理非线性数据

回到 SVM 的对偶问题

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0. \tag{12}$$

根据我们之前的推导，这里的 $\boldsymbol{Q}$ 矩阵为 $Q_{ij} = y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j$.

如果将其中的 $\boldsymbol{x}$ 替换为 $\phi(\boldsymbol{x})$，也就是让 $Q_{ij} = y_i y_j \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j)$，就相当于:

1. 先对所有数据施加非线性映射 $\phi(\cdot)$，
2. 再用 $\phi(\boldsymbol{x}_i)$ 作为输入来训练线性 SVM.

再进一步，定义一个函数 $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^\top \phi(\boldsymbol{x}')$，则有 $Q_{ij} = y_i y_j \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)$. 这就是核 SVM 的初步思想.

如何处理非线性数据

回到 SVM 的对偶问题

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \mathbf{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0. \tag{12}$$

根据我们之前的推导，这里的 $\boldsymbol{Q}$ 矩阵为 $Q_{ij} = y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j$.

如果将其中的 $\boldsymbol{x}$ 替换为 $\phi(\boldsymbol{x})$，也就是让 $Q_{ij} = y_i y_j \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j)$，就相当于：

1. 先对所有数据施加非线性映射 $\phi(\cdot)$，
2. 再用 $\phi(\boldsymbol{x}_i)$ 作为输入来训练线性 SVM.

再进一步，定义一个函数 $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^\top \phi(\boldsymbol{x}')$，则有 $Q_{ij} = y_i y_j \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)$.
这就是核 SVM 的初步思想.

深圳大学 SHENZHEN UNIVERSITY

核技巧[2]

定义 (正定对称核)

定义函数 $\mathcal{K}:\mathbb{R}^d\times\mathbb{R}^d\mapsto\mathbb{R}$. 若对任意在 $\mathbb{R}^d$ 上的数据集: $\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n\}$, 矩阵 $\boldsymbol{K}\in\mathbb{R}^{n\times n}$ 满足 $K_{ij}=\mathcal{K}(\boldsymbol{x}_i,\boldsymbol{x}_j)$, 且 $\mathcal{K}$ 是对称正定的, 则称 $\mathcal{K}$ 是正定对称核.

对任意正定对称核, 存在唯一的特征空间 $\mathcal{H}$ 以及映射 $\phi:\mathbb{R}^d\mapsto\mathcal{H}$, 使得

$$\mathcal{K}(\boldsymbol{x},\boldsymbol{x}')=\phi(\boldsymbol{x})^\top\phi(\boldsymbol{x}'),\quad\forall\boldsymbol{x},\boldsymbol{x}'.\tag{13}$$

简单来说，核方法的精髓是:

- 一个正定对称核 $\mathcal{K}$ 确定了一个非线性映射 $\phi(\cdot)$;
- 在核方法中不需要自己设计 $\phi(\cdot)$，只需要知道核函数 $\mathcal{K}$.
- $\phi(x)$ 一般非常复杂，并且维度很高，一般是无法计算的 (intractable)，但计算映射后的内积只需要 (13)，往往只需要 $O(d)$ 时间.
- 一个常用核函数，高斯核: $\mathcal{K}(x,x')=\exp(-\gamma\|x-x'\|^2)$.

[2]敬请期待未来的核方法专题...

核技巧[2]

定义 (正定对称核)

定义函数 $\mathcal{K}:\mathbb{R}^d\times\mathbb{R}^d\mapsto\mathbb{R}$. 若对任意在 $\mathbb{R}^d$ 上的数据集: $\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n\}$, 矩阵 $\boldsymbol{K}\in\mathbb{R}^{n\times n}$ 满足 $K_{ij}=\mathcal{K}(\boldsymbol{x}_i,\boldsymbol{x}_j)$, 且 $\mathcal{K}$ 是对称正定的, 则称 $\mathcal{K}$ 是正定对称核.

对任意正定对称核, 存在唯一的特征空间 $\mathcal{H}$ 以及映射 $\phi:\mathbb{R}^d\mapsto\mathcal{H}$, 使得

$$\mathcal{K}(\boldsymbol{x},\boldsymbol{x}')=\phi(\boldsymbol{x})^\top\phi(\boldsymbol{x}'),\quad\forall\boldsymbol{x},\boldsymbol{x}'.\tag{13}$$

简单来说，核方法的精髓是：

- 一个正定对称核 $\mathcal{K}$ 确定了一个非线性映射 $\phi(\cdot)$；
- 在核方法中不需要自己设计 $\phi(\cdot)$，只需要知道核函数 $\mathcal{K}$.
- $\phi(\boldsymbol{x})$ 一般非常复杂，并且维度很高，一般是无法计算的 (intractable)，但计算映射后的内积只需要 (13)，往往只需要 $O(d)$ 时间.
- 一个常用核函数，高斯核: $\mathcal{K}(\boldsymbol{x},\boldsymbol{x}')=\exp(-\gamma\|\boldsymbol{x}-\boldsymbol{x}\|^2)$.

[2] 敬请期待未来的核方法专题...

核技巧[2]

定义 (正定对称核)

定义函数 $\mathcal{K}:\mathbb{R}^d\times\mathbb{R}^d\mapsto\mathbb{R}$. 若对任意在 $\mathbb{R}^d$ 上的数据集: $\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n\}$, 矩阵 $\boldsymbol{K}\in\mathbb{R}^{n\times n}$ 满足 $K_{ij}=\mathcal{K}(\boldsymbol{x}_i,\boldsymbol{x}_j)$, 且 $\mathcal{K}$ 是对称正定的, 则称 $\mathcal{K}$ 是正定对称核.

对任意正定对称核, 存在唯一的特征空间 $\mathcal{H}$ 以及映射 $\phi:\mathbb{R}^d\mapsto\mathcal{H}$, 使得

$$\mathcal{K}(\boldsymbol{x},\boldsymbol{x}')=\phi(\boldsymbol{x})^\top\phi(\boldsymbol{x}'),\quad\forall\boldsymbol{x},\boldsymbol{x}'.\tag{13}$$

简单来说，核方法的精髓是:

- 一个正定对称核 $\mathcal{K}$ 确定了一个非线性映射 $\phi(\cdot)$;
- 在核方法中不需要自己设计 $\phi(\cdot)$，只需要知道核函数 $\mathcal{K}$.
- $\phi(x)$ 一般非常复杂，并且维度很高，一般是无法计算的 (intractable)，但计算映射后的内积只需要 (13)，往往只需要 $O(d)$ 时间.
- 一个常用核函数，高斯核: $\mathcal{K}(x,x')=\exp(-\gamma\|x-x\|^2)$.

深圳大学 SHENZHEN UNIVERSITY

[2]敬请期待未来的核方法专题...

核技巧[2]

定义 (正定对称核)

定义函数 $\mathcal{K}:\mathbb{R}^d\times\mathbb{R}^d\mapsto\mathbb{R}$. 若对任意在 $\mathbb{R}^d$ 上的数据集: $\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n\}$, 矩阵 $\boldsymbol{K}\in\mathbb{R}^{n\times n}$ 满足 $K_{ij}=\mathcal{K}(\boldsymbol{x}_i,\boldsymbol{x}_j)$, 且 $\mathcal{K}$ 是对称正定的, 则称 $\mathcal{K}$ 是正定对称核.

对任意正定对称核, 存在唯一的特征空间 $\mathcal{H}$ 以及映射 $\phi:\mathbb{R}^d\mapsto\mathcal{H}$, 使得

$$\mathcal{K}(\boldsymbol{x},\boldsymbol{x}')=\phi(\boldsymbol{x})^\top\phi(\boldsymbol{x}'),\quad\forall\boldsymbol{x},\boldsymbol{x}'.\tag{13}$$

简单来说，核方法的精髓是：

- 一个正定对称核 $\mathcal{K}$ 确定了一个非线性映射 $\phi(\cdot)$；
- 在核方法中不需要自己设计 $\phi(\cdot)$，只需要知道核函数 $\mathcal{K}$.
- $\phi(\boldsymbol{x})$ 一般非常复杂，并且维度很高，一般是无法计算的 (intractable)，但计算映射后的内积只需要 (13)，往往只需要 $O(d)$ 时间.
- 一个常用核函数，高斯核: $\mathcal{K}(\boldsymbol{x},\boldsymbol{x}')=\exp(-\gamma\|\boldsymbol{x}-\boldsymbol{x}\|^2)$.

深圳大学 SHENZHEN UNIVERSITY

[2]敬请期待未来的核方法专题...

核技巧[2]

定义 (正定对称核)

定义函数 $\mathcal{K}:\mathbb{R}^d\times\mathbb{R}^d\mapsto\mathbb{R}$. 若对任意在 $\mathbb{R}^d$ 上的数据集: $\{\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n\}$, 矩阵 $\boldsymbol{K}\in\mathbb{R}^{n\times n}$ 满足 $K_{ij}=\mathcal{K}(\boldsymbol{x}_i,\boldsymbol{x}_j)$, 且 $\mathcal{K}$ 是对称正定的, 则称 $\mathcal{K}$ 是正定对称核.

对任意正定对称核, 存在唯一的特征空间 $\mathcal{H}$ 以及映射 $\phi:\mathbb{R}^d\mapsto\mathcal{H}$, 使得

$$\mathcal{K}(\boldsymbol{x},\boldsymbol{x}')=\phi(\boldsymbol{x})^\top\phi(\boldsymbol{x}'),\quad\forall\boldsymbol{x},\boldsymbol{x}'. \tag{13}$$

简单来说，核方法的精髓是:

- 一个正定对称核 $\mathcal{K}$ 确定了一个非线性映射 $\phi(\cdot)$;
- 在核方法中不需要自己设计 $\phi(\cdot)$，只需要知道核函数 $\mathcal{K}$.
- $\phi(\boldsymbol{x})$ 一般非常复杂，并且维度很高，一般是无法计算的 (intractable)，但计算映射后的内积只需要 (13)，往往只需要 $O(d)$ 时间.
- 一个常用核函数，高斯核: $\mathcal{K}(\boldsymbol{x},\boldsymbol{x}')=\exp(-\gamma\|\boldsymbol{x}-\boldsymbol{x}\|^2)$.

[2] 敬请期待未来的核方法专题...

核技巧[2]

定义 (正定对称核)

定义函数 $\mathcal{K}: \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$. 若对任意在 $\mathbb{R}^d$ 上的数据集: $\{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n\}$, 矩阵 $\boldsymbol{K} \in \mathbb{R}^{n\times n}$ 满足 $K_{ij} = \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)$, 且 $\mathcal{K}$ 是对称正定的, 则称 $\mathcal{K}$ 是正定对称核.

对任意正定对称核, 存在唯一的特征空间 $\mathcal{H}$ 以及映射 $\phi: \mathbb{R}^d \mapsto \mathcal{H}$, 使得

$$\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^\top \phi(\boldsymbol{x}'), \quad \forall \boldsymbol{x}, \boldsymbol{x}'. \tag{13}$$

简单来说，核方法的精髓是:

- 一个正定对称核 $\mathcal{K}$ 确定了一个非线性映射 $\phi(\cdot)$;
- 在核方法中不需要自己设计 $\phi(\cdot)$，只需要知道核函数 $\mathcal{K}$.
- $\phi(\boldsymbol{x})$ 一般非常复杂，并且维度很高，一般是无法计算的 (intractable)，但计算映射后的内积只需要 (13)，往往只需要 $O(d)$ 时间.
- 一个常用核函数，高斯核: $\mathcal{K}(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\gamma\|\boldsymbol{x} - \boldsymbol{x}\|^2)$.

[2] 敬请期待未来的核方法专题...

核 SVM (非线性 SVM)

对偶问题与线性 SVM 一致，只是将 $\boldsymbol{Q}$ 矩阵换成 $Q_{ij} = y_i y_j \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j)$:

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \boldsymbol{1}^\top \boldsymbol{\alpha}, \quad \text{s.t. } 0 \le \alpha_i \le c,\ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0.$$

注意到，此时 $\boldsymbol{w}^\star$ 的表达式变成了:

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

而对新样本 $\boldsymbol{x}$ 的预测为

$$\begin{aligned} h(\boldsymbol{x}) &= \operatorname{sign}(\phi(\boldsymbol{x})^\top \boldsymbol{w}^\star + b^\star) \\ &= \operatorname{sign}\left(\sum_{i:\alpha_i \neq 0} \alpha_i^\star y_i \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j) + b^\star\right) = \operatorname{sign}\left(\sum_{i:\alpha_i \neq 0} \alpha_i^\star y_i \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}) + b^\star\right). \end{aligned}$$

目录

实际应用中的 SVM

- LIBSVM [CL11]:
  - Good: 用 C++ 基于 SMO 算法实现了核SVM，被收编进 sklearn 的开源库.
  - Bad: SMO 算法老矣，对大数据集的训练力不从心.
- LIBLINEAR [FCH+08]:
  - Good: 对线性 SVM 进行了专门优化，大幅提高训练速度，适用于大数据.
  - Bad: C++ 代码中的数据存储结构不够好，处理稠密数据效率低 (LIBSVM 有类似问题).
- ThunderSVM [WSL+18]:
  - Good: 由 C++ 和 CUDA 实现的 SVM，相当于 GPU 加速的 LIBSVM.
  - Bad: 大规模数据仍然比较慢.

深圳大学 SHENZHEN UNIVERSITY

关于 SVM 的更多阅读文献

- (1999 NPL) 最小二乘 SVM [SV99];
- (2001 JMLR) 多分类 CS-SVM [CS01];
- (2006 JMLR) 针对半监督问题的 Lap-SVM [BNS06];
- (2007 PAMI) 孪生 SVM [KC+07];

近年来我们的工作:

- (2023 TCYB) 鲁棒流形孪生 SVM [ZLKS23];
- (2023 TCYB) 最大间隔嵌入 SVM [LCZ+23];
- (2024 CAAI) 低秩嵌入 SVM [LLK24];
- (2024 INS) 联合最优多分类 SVM [LLZ+24]
- (2025 PAMI) 最优鉴别特征 SVM [ZLKY25];
- (coming...) NPSVC++ (https://arxiv.org/abs/2402.06010).

结束

谢谢!

https://github.com/Apple-Zhang/SVM-Intro

Contact with Apple!
Email: apple_zjh@163.com

参考文献 I

[BNS06] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani.
Manifold regularization: A geometric framework for learning from labeled and unlabeled examples.
Journal of Machine Learning Research, 7(11), 2006.

[CL11] Chih-Chung Chang and Chih-Jen Lin.
Libsvm: a library for support vector machines.
ACM Transactions on Intelligent Systems and Technology (TIST), 2(3):1–27, 2011.

[CS01] Koby Crammer and Yoram Singer.
On the algorithmic implementation of multiclass kernel-based vector machines.
Journal of Machine Learning Research, 2(Dec):265–292, 2001.

[CV95] Corinna Cortes and Vladimir Vapnik.
Support-vector networks.
Machine Learning, 20(3):273–297, 1995.

[FCH+08] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin.
Liblinear: A library for large linear classification.
the Journal of Machine Learning Research, 9:1871–1874, 2008.

[KC+07] Reshma Khemchandani, Suresh Chandra, et al.
Twin support vector machines for pattern classification.
IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(5):905–910, 2007.

[LCZ+23] Zhihui Lai, Xi Chen, Junhong Zhang, Heng Kong, and Jiajun Wen.
Maximal margin support vector machine for feature representation and classification.
IEEE Transactions on Cybernetics, 53(10):6700–6713, 2023.

深圳大学 SHENZHEN UNIVERSITY

参考文献 II

[LLK24] Guangfei Liang, Zhihui Lai, and Heng Kong.
Support vector machine with discriminative low-rank embedding.
CAAI Transactions on Intelligence Technology, 9(5):1249–1262, 2024.

[LLZ+24] Zhihui Lai, Guangfei Liang, Jie Zhou, Heng Kong, and Yuwu Lu.
A joint learning framework for optimal feature extraction and multi-class svm.
Information Sciences, 671:120656, 2024.

[SV99] Johan AK Suykens and Joos Vandewalle.
Least squares support vector machine classifiers.
Neural Processing Letters, 9(3):293–300, 1999.

[WSL+18] Zeyi Wen, Jiashuai Shi, Qinbin Li, Bingsheng He, and Jian Chen.
Thundersvm: A fast svm library on gpus and cpus.
Journal of Machine Learning Research, 19(21):1–5, 2018.

[ZLKS23] Junhong Zhang, Zhihui Lai, Heng Kong, and Linlin Shen.
Robust twin bounded support vector classifier with manifold regularization.
IEEE Transactions on Cybernetics, 53(8):5135–5150, 2023.

[ZLKY25] Junhong Zhang, Zhihui Lai, Heng Kong, and Jian Yang.
Learning The Optimal Discriminant SVM with Feature Extraction.
IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 1–15, 2025.

深圳大学 SHENZHEN UNIVERSITY