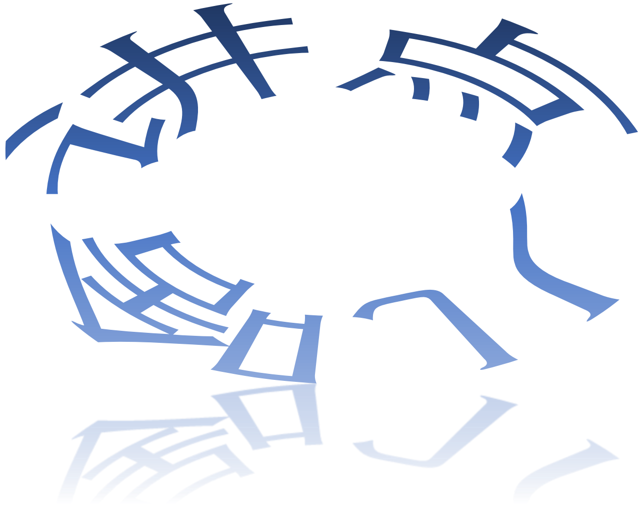
第5章 支持向量机

Support Vector Machine

电子工程学院、人工智能学院
college of Electronic Engineering , college of Artificial Intelligence



5.1 支持向量

5.2 对偶问题

5.3 核函数

5.4 软间隔与正则化

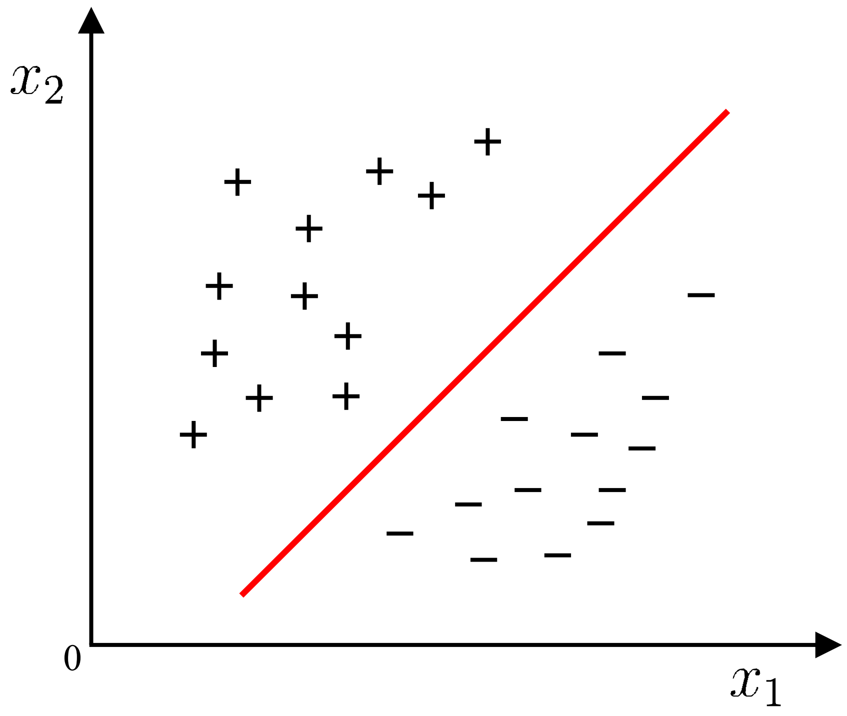
5.5 合页损失函数

5.6 对偶问题II

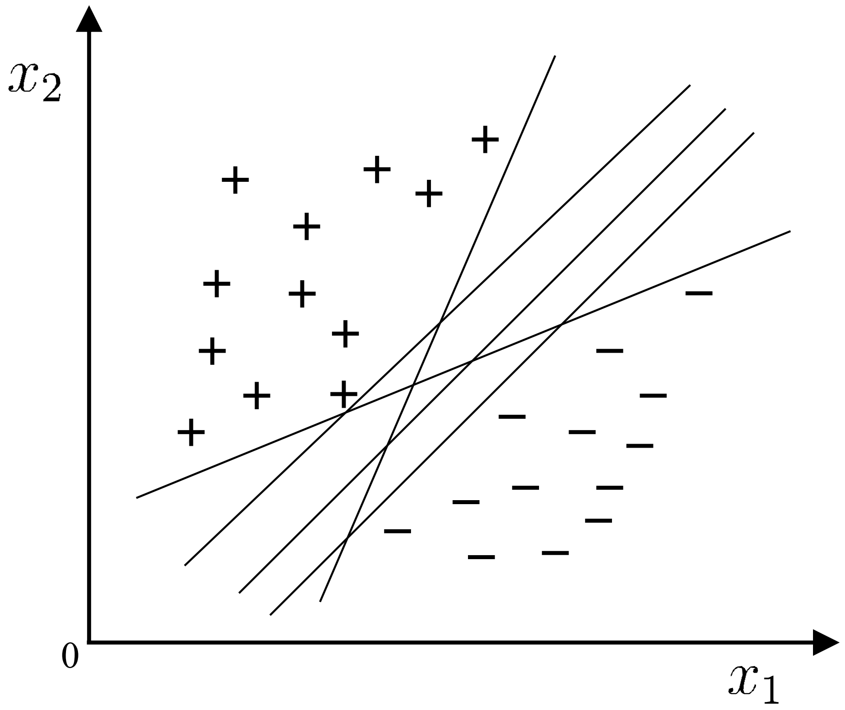
5.7 支持向量回归

5.1 支持向量(Support Vector)

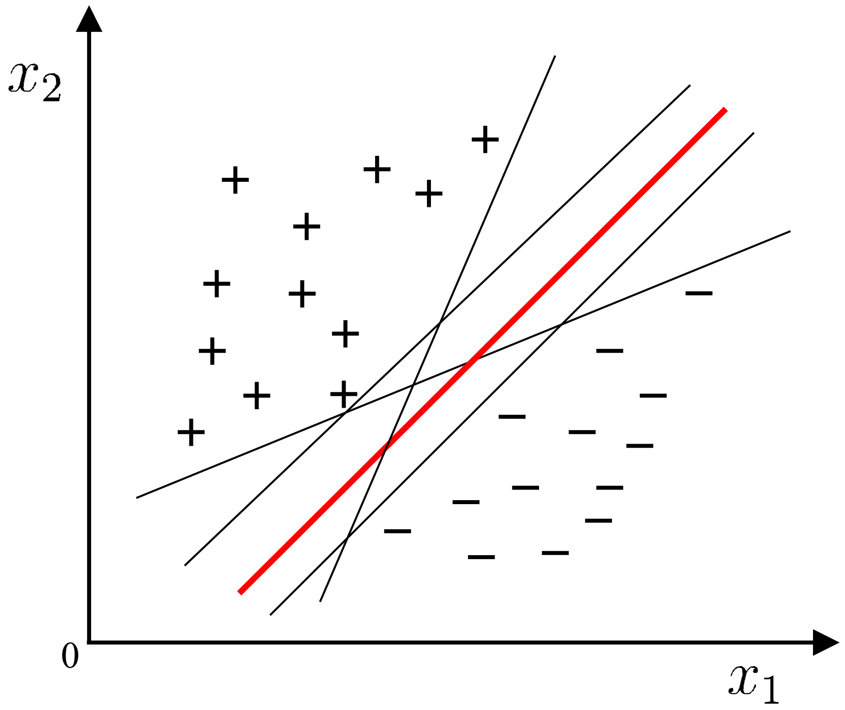
线性模型：在样本空间中寻找一个超平面, 将不同类别的样本分开.



-Q: 将训练样本分开的超平面可能有很多, 哪一个好呢?



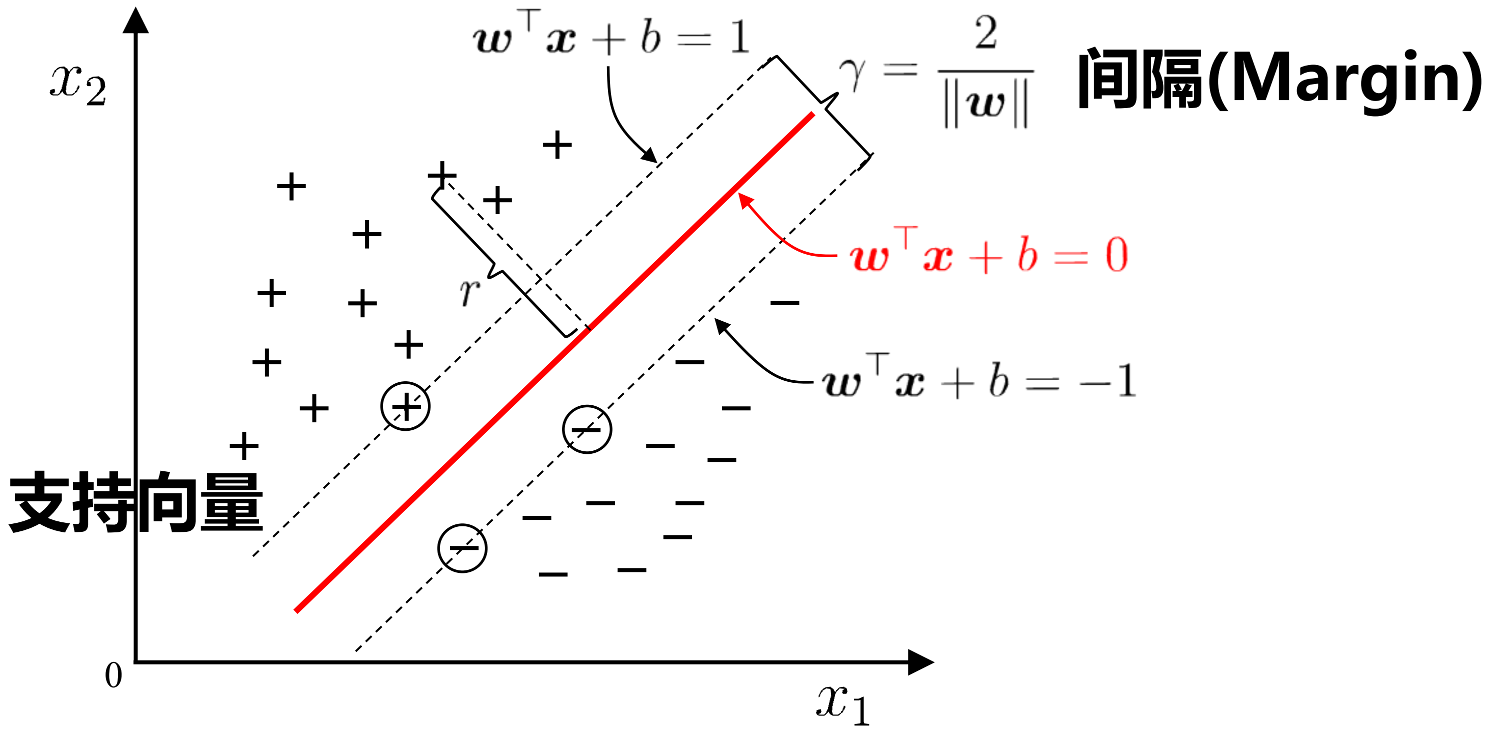
-Q: 将训练样本分开的超平面可能有很多, 哪一个好呢?



-A:应选择”正中间”的超平面, 容忍性最好, 鲁棒性最高, 泛化能力最强.

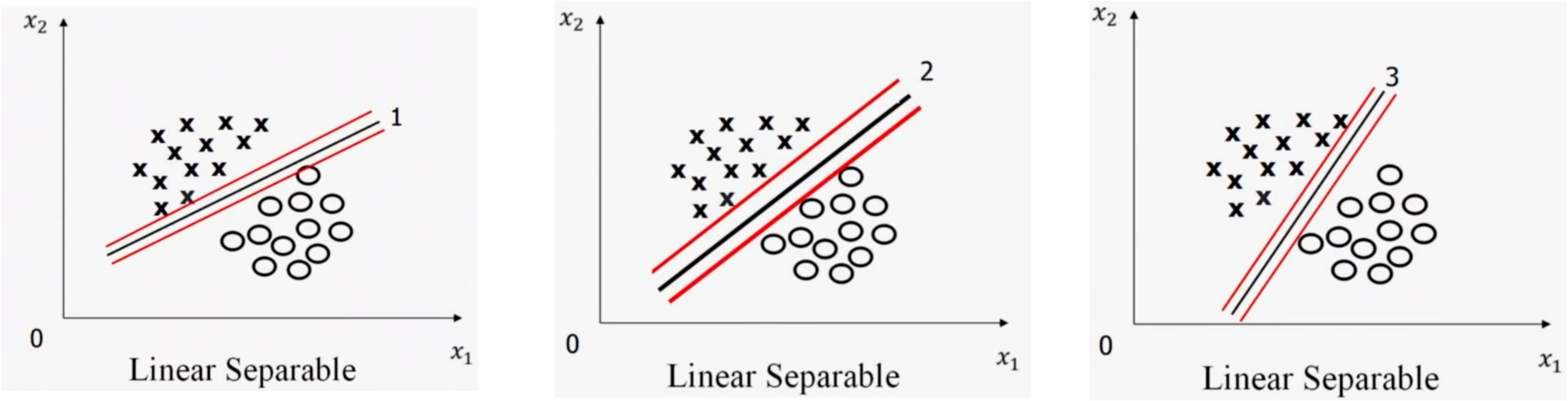
5.1.1 支持向量

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



Vladimir Vapnik

间隔（Margins）最大



直线能分开两类
寻找间隔最大的直线
直线在间隔的正中间

$$\gamma = \frac{2}{\|\boldsymbol{w}\|}$$

超平面能分开两类
寻找间隔最大的超平面
超平面在间隔的正中间

5.1.2 支持向量机最优化问题

- 最大间隔: 寻找参数 w 和b, 使得γ 最大.

$$\underset{w,b}{argmax} \frac{1}{||w||}$$

$$s.t.\ y_i(w^T x_i + b) \geq 1,\ i = 1,2,..,m$$

⇩

$$\underset{w,b}{argmin}\ \frac{1}{2}||w||^2$$

凸二次规划

(convex quadratic programming)

$$s.t.\ y_i(w^T x_i + b) \geq 1,\ i = 1,2,..,m$$

(6.6)

点到平面的距离：

$$w^T x + b = 0$$

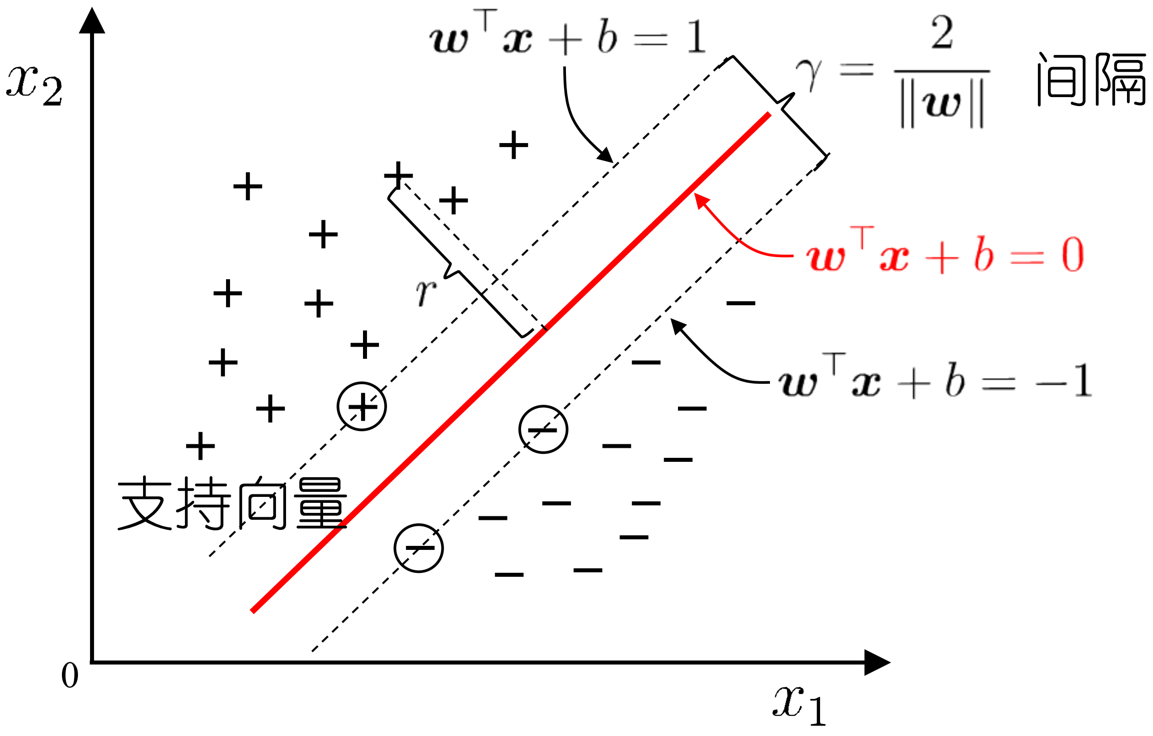
二维空间点 (x, y)到直线 $Ax + By + C = 0$的距离公式是：

$$\frac{|Ax + By + C|}{\sqrt{A^2 + B^2}}$$

扩展到 n 维空间后，点 $x = (x_1, x_2 \ldots x_n)$ 到超平面

$w^T x + b = 0$ 的距离为：$d = \dfrac{|w^T x + b|}{||w||}$

其中 $||w|| = \sqrt{w_1^2 + \cdots w_n^2}$

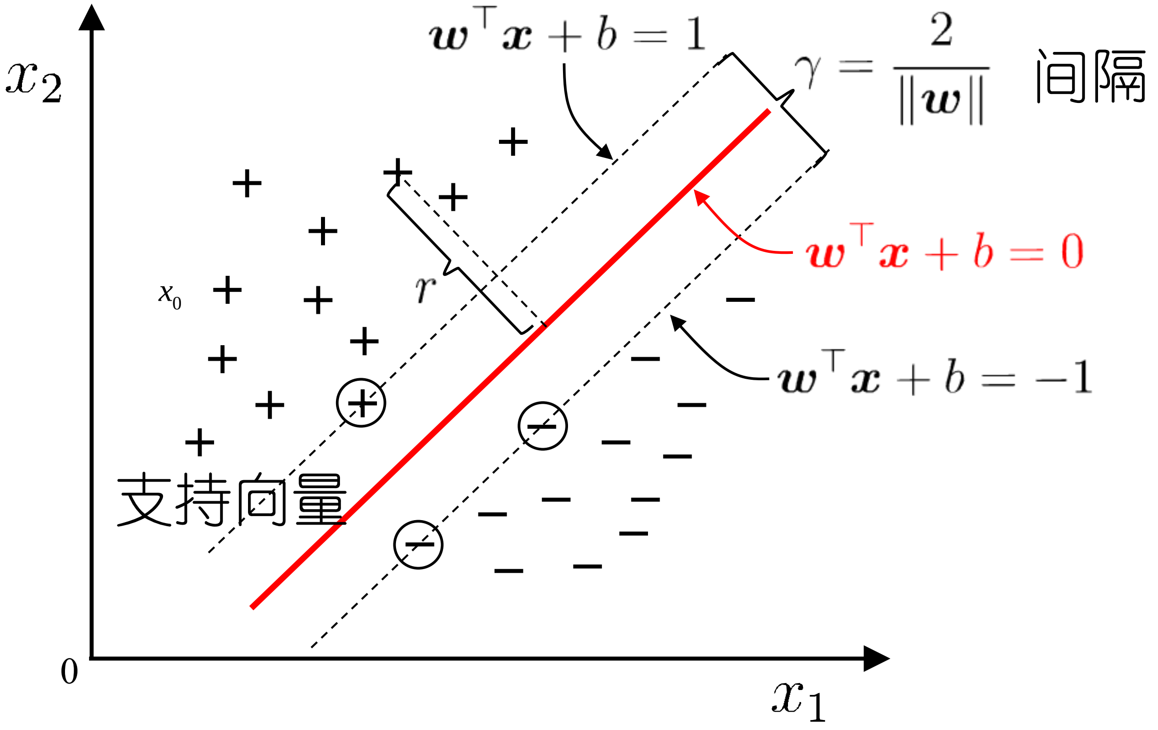


决策超平面：

$w^T x + b = 0$ 与 $(aw^T)x + ab = 0$ $(a \neq 0)$ 是同一个超平面

$(w, b) \longrightarrow (aw, ab)$

支持向量决定的超平面：



支持向量x_i $\begin{cases} w^T x_i + b = 1 & y_i=+1 \\ w^T x_i + b = -1 & y_i=-1 \end{cases}$

非支持向量x_i $\begin{cases} w^T x_i + b > 1 & y_i=+1 \\ w^T x_i + b < -1 & y_i=-1 \end{cases}$

支持向量 x 到超平面的距离 $d = \frac{|w^T x + b|}{||w||} = \frac{1}{||w||}$

- 支持向量 x 到超平面的距离

$$d = \frac{|w^T x + b|}{||w||} = \frac{1}{||w||}$$

最大化 $\frac{1}{||w||}$，即最小化$||w||$ $\Rightarrow$ $\underset{w,b}{argmin}\ \frac{1}{2}||w||^2$

- 正确分类样本 x 到超平面的距离

$$w^T x_i + b \geq 1 \quad y_i = +1$$

$$w^T x_i + b \leq -1 \quad y_i = -1$$

$$\Downarrow$$

$$s.t.\ y_i(w^T x_i + b) \geq 1,\ i = 1,2,..,m$$

$$\boxed{\begin{gathered}\underset{w,b}{argmin}\ \frac{1}{2}||w||^2 \\ s.t.\ y_i(w^T x_i + b) \geq 1,\ i = 1,2,..,m\end{gathered}}$$

5.2 对偶问题（dual problem)

$$\underset{w,b}{argmin}\ \frac{1}{2}||w||^2$$

$$s.t.\ y_i(w^T x_i + b) \geq 1,\ i = 1,2,..,m$$

(6.6)

拉格朗日乘子法

- 第一步：引入拉格朗日乘子$\alpha_i \geq 0$得到拉格朗日函数

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + \sum_{i=1}^{m} \alpha_i \left(1 - y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b)\right)$$

- 第二步：令$L(\boldsymbol{w}, b, \boldsymbol{\alpha})$ 对$\boldsymbol{w}$ 和b 的偏导为零可得

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i\ ,\quad 0 = \sum_{i=1}^{m} \alpha_i y_i$$

- 第三步：回代可得

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j$$

$$\text{s.t.} \sum_{i=1}^{m} \alpha_i y_i = 0\ ,\quad \alpha_i \geqslant 0\ ,\quad i = 1, 2, \ldots, m$$

解的特性——稀疏性

最终模型： $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i^\top \boldsymbol{x} + b$

KKT条件：

$$\begin{cases} \alpha_i \geq 0\,; \\ 1 - y_i f(\boldsymbol{x}_i) \leq 0\,; \\ \alpha_i \left(1 - y_i f(\boldsymbol{x}_i)\right) = 0\,. \end{cases} \Longrightarrow$$

必有 $\alpha_i = 0$ 或

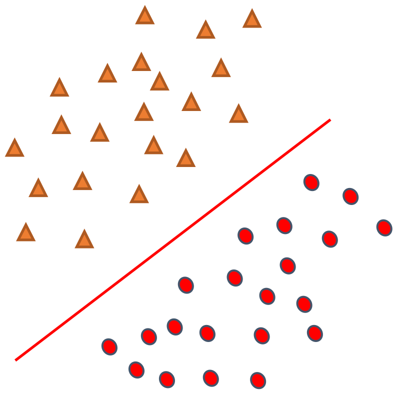
$y_i f(\boldsymbol{x}_i) = 1$

解的稀疏性: 训练完成后, 最终模型仅与支持向量有关

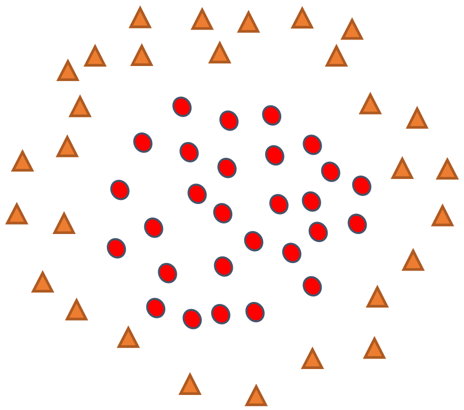
支持向量机(Support Vector Machine, SVM) 因此而得名

5.3 核函数（Kernel Function）

5.3.1 数据线性可分与线性不可分



线性可分



线性不可分

定义线性可分

$$D=\{(x_1,y_1),(x_2,y_2),......(x_m,y_m)\}$$

$$y\in\{+1,-1\}$$

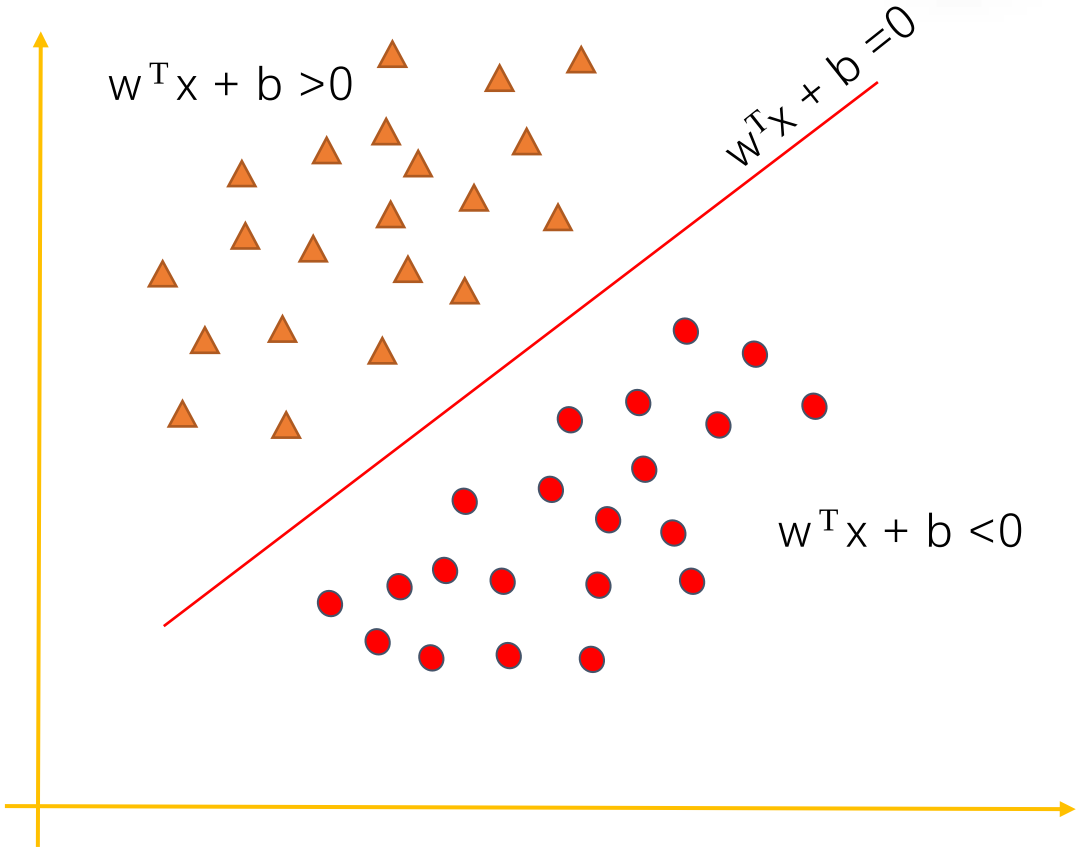
- 线性分类器$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$
- 若存在w，使得：

所有满足$f(\mathbf{x}) < 0$ 的点，其对应的y等于-1

所有满足$f(\mathbf{x}) > 0$ 的点，其对应的y等于1

则数据线性可分

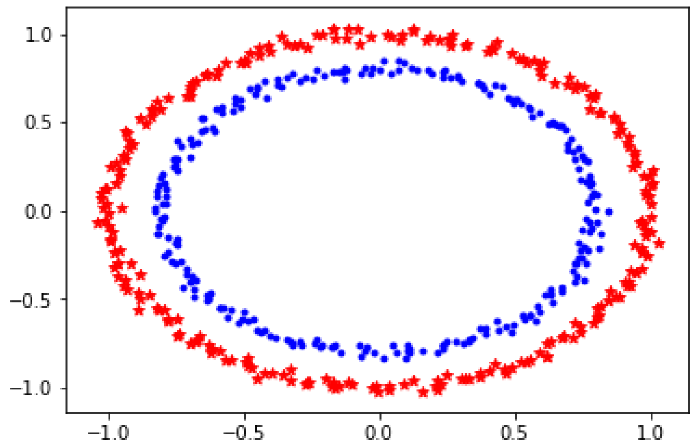
- 线性判别函数$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b = 0$，$\mathbf{x}$是位于超平面面上的点

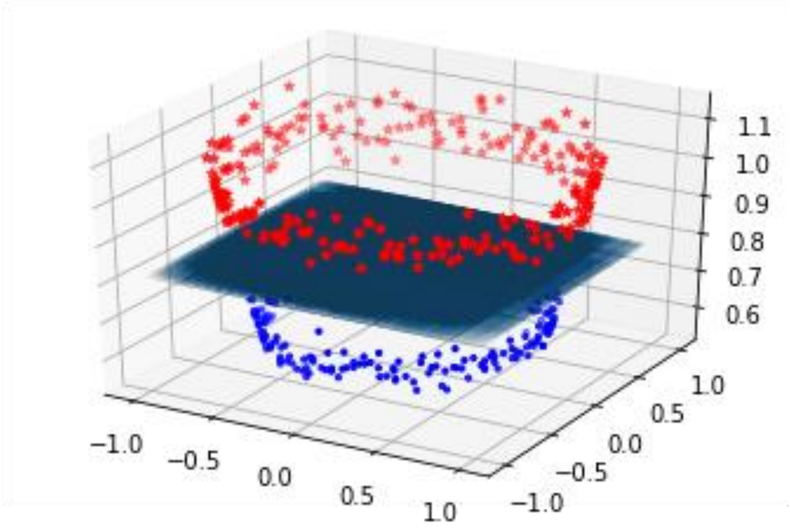


超平面：$w^T x + b = 0$

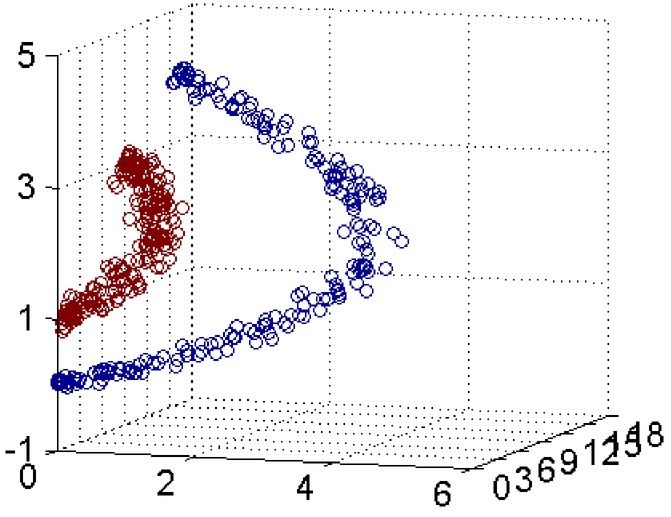
5.3.2 特征空间映射

$$(z_1, z_2, z_3) = (x, y, x^2 + y^2)$$

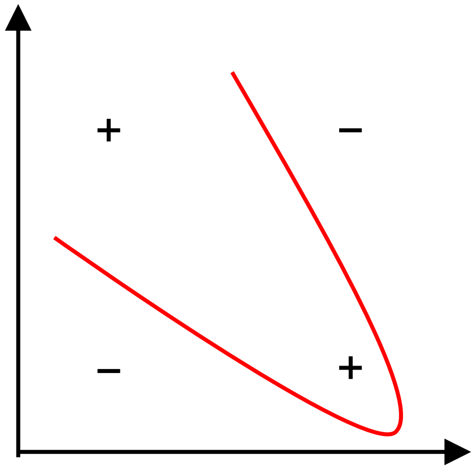




将原始空间映射到一个更高维特征空间，使得在这个特征空间数据线性可分。

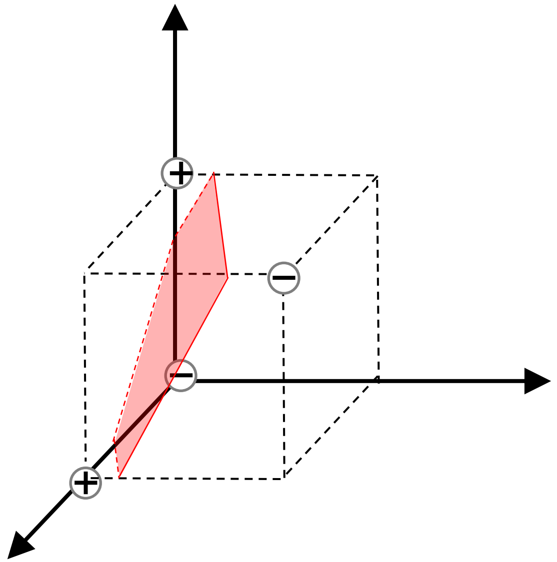


高维下线性可分



XOR数据集

$$\boldsymbol{x} \mapsto \phi(\boldsymbol{x})$$



如果原始空间是有限维(属性数有限)，那么一定存在一个高维特征空间使样本线性可分

5.3.3 高维空间中的最优化问题

设样本 $\boldsymbol{x}$ 映射后的向量为$\phi(\boldsymbol{x})$, 划分超平面为$f(\boldsymbol{x}) = \boldsymbol{w}^\top \phi(\boldsymbol{x}) + b$

原始问题

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

对偶问题

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

只以内积形式出现

预测

$$f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}} \phi(\boldsymbol{x}) + b = \sum_{i=1}^{m} \alpha_i y_i \boxed{\phi(\boldsymbol{x}_i)^{\mathrm{T}} \phi(\boldsymbol{x})} + b$$

5.3.4 核函数 (Kernel Function)

基本思路：设计核函数

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^{\mathrm{T}} \phi(\boldsymbol{x}_j)$$

绕过显式考虑特征映射、以及计算高维内积的困难

核技巧（Kernel Trick)

“核函数选择”成为决定支持向量机性能的关键！

表6.1 常用核函数

名称	表达式	参数
线性核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^\top \boldsymbol{x}_j$	
多项式核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^\top \boldsymbol{x}_j)^d$	$d \geq 1$为多项式的次数
高斯核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\Vert\boldsymbol{x}_i - \boldsymbol{x}_j\Vert^2}{2\delta^2}\right)$	$\delta > 0$为高斯核的带宽(width)
拉普拉斯核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\Vert\boldsymbol{x}_i - \boldsymbol{x}_j\Vert}{\delta}\right)$	$\delta > 0$
Sigmoid核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \tanh(\beta \boldsymbol{x}_i^\top \boldsymbol{x}_j + \theta)$	tanh为双曲正切函数，$\beta > 0$, $\theta < 0$

例: 2项式核

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$\phi(x) = \begin{bmatrix} {x_1}^2 \\ \sqrt{2}x_1x_2 \\ {x_2}^2 \end{bmatrix}$$

$$K(x,z) = \phi(x)\cdot\phi(z) = \begin{bmatrix} {x_1}^2 \\ \sqrt{2}x_1x_2 \\ {x_2}^2 \end{bmatrix} \cdot \begin{bmatrix} {z_1}^2 \\ \sqrt{2}z_1z_2 \\ {z_2}^2 \end{bmatrix}$$

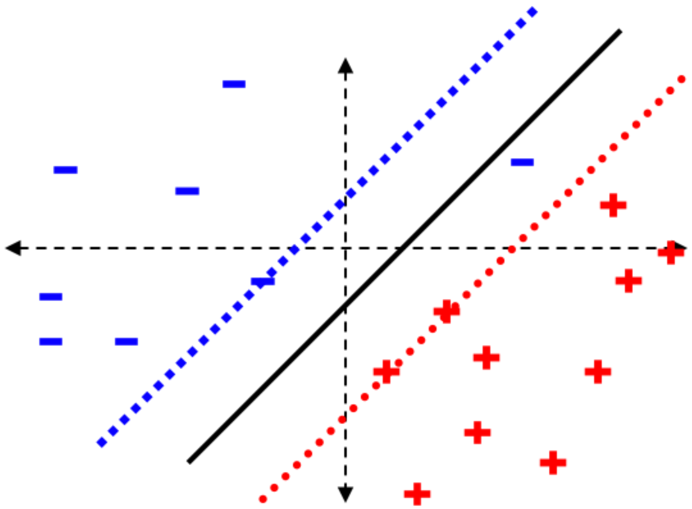
$$= {x_1}^2{z_1}^2 + 2x_1x_2z_1z_2 + {x_2}^2{z_2}^2$$

$$= (x_1z_1 + x_2z_2)^2 = \left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \cdot \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}\right)^2$$

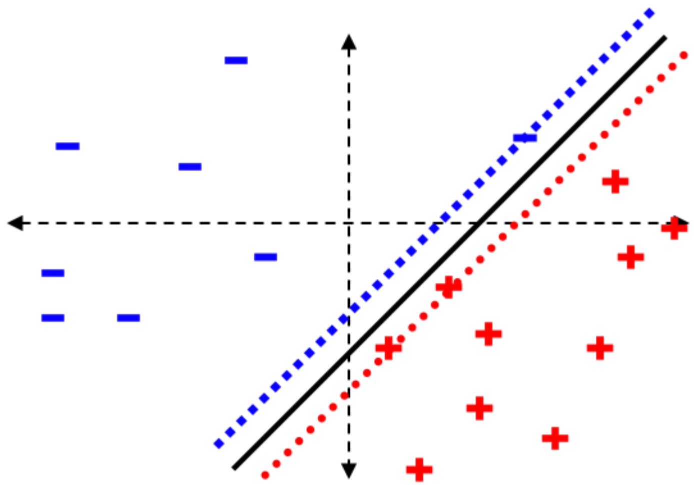
$$= (x\cdot z)^2$$

5.4 软间隔与正则化

5.4.1 软间隔



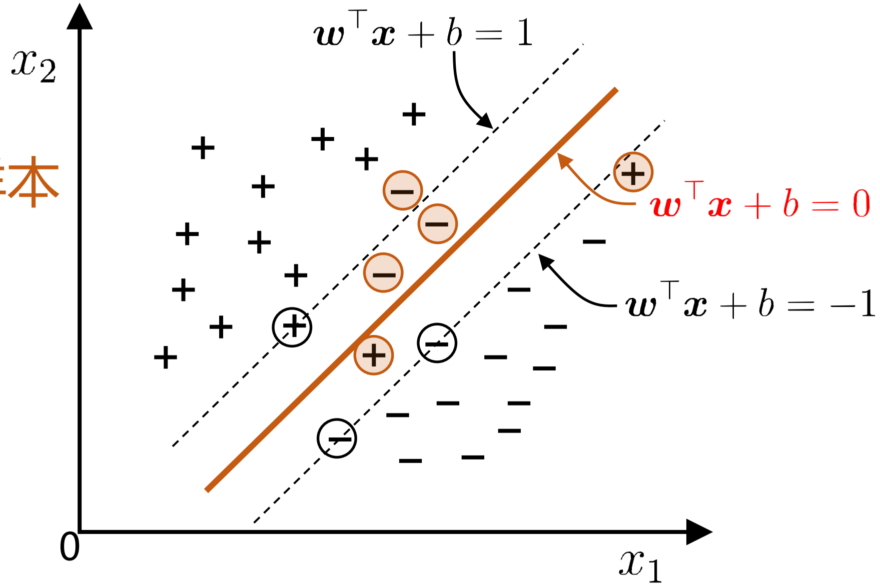
少量样本被错分，但间隔大



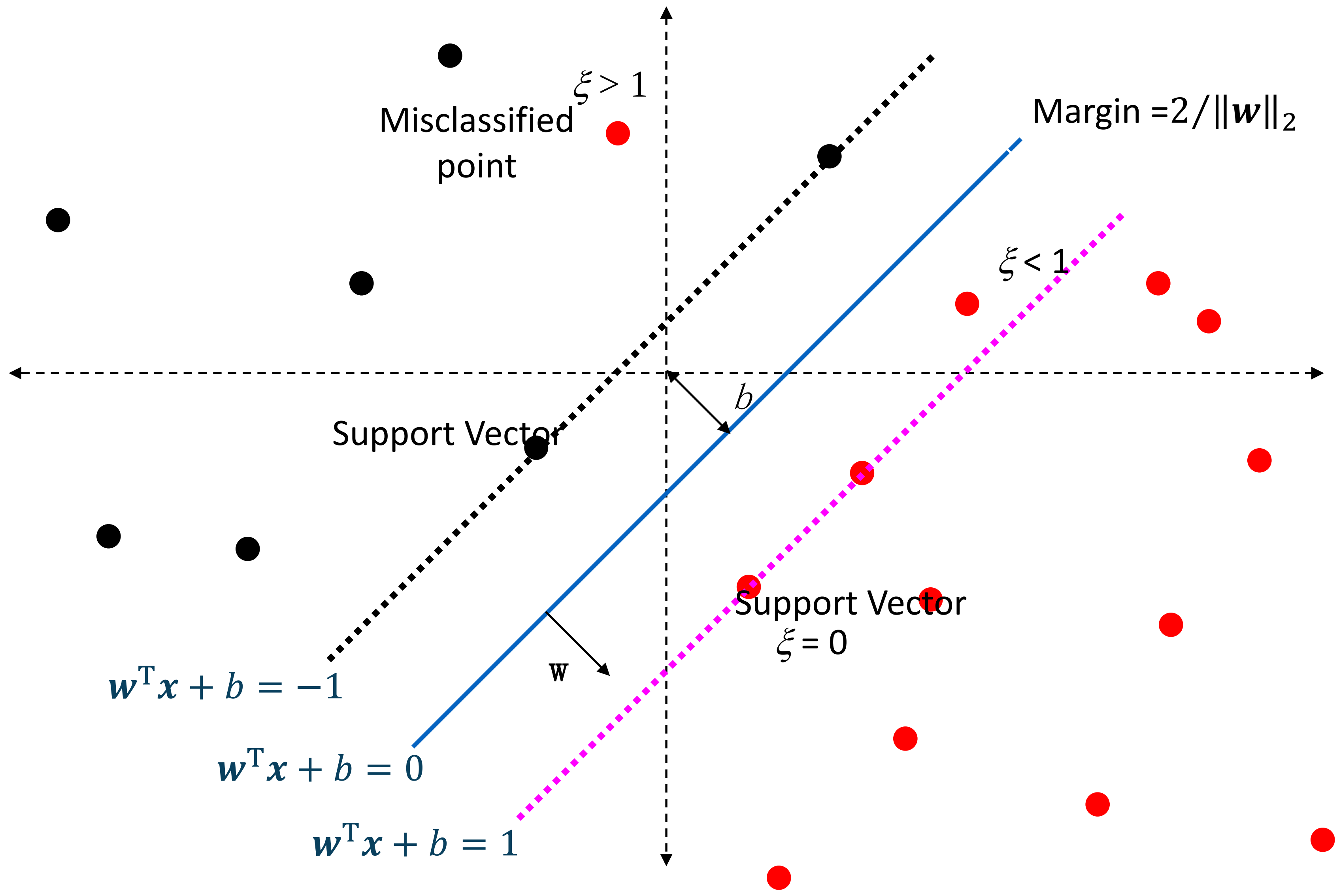
样本被完全分对，但间隔小

引入软间隔 (Soft Margin), 允许在一些样本上不满足约束，这些样本称为**松弛变量**，记作ξ

不满足约束的样本



松弛变量ξ



$\xi > 1$
Misclassified point
Margin $=2/\|\boldsymbol{w}\|_2$
$\xi < 1$
b
Support Vector
Support Vector
$\xi = 0$
w
$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b = -1$
$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b = 0$
$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b = 1$

5.4.2 C-SVM

若数据线性不可分，则可以引入松弛变量(slack variable)$\xi \geq 0$，使函数间隔加上“**松弛变量**”大于等于1

$$y_i(w^T x_i + b) \geq 1 - \xi_i$$

则软间隔最大化SVM（C-SVM）的**目标函数**

$$J(\boldsymbol{w}, b, C) = C\sum_{i=1}^{m} \xi_i + \frac{1}{2}\|\boldsymbol{w}\|_2^2$$

$$\text{s.t. } y_i\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \qquad i = 1,2,\dots,m$$

$$\xi_i \geq 0, \quad i = 1,2,\dots,m$$

➢ C-SVM目标函数

$$\underset{w,b}{argmin}\ \frac{1}{2}||w||^2$$

$$s.t.\ y_i(w^Tx_i+b)\geq 1,\ i=1,2,\ldots,m$$

$$J(\boldsymbol{w},b,C)=\underset{w,b,\xi_i}{argmin}\ C\sum_{i=1}^{m}\xi_i+\frac{1}{2}\|\boldsymbol{w}\|_2^2$$

$$\text{s.t. } y_i\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i+b\right)\geq 1-\xi_i,\qquad i=1,2,\ldots,m \qquad \textbf{(6.35)}$$

$$\xi_i\geq 0,\quad i=1,2,\ldots,m$$

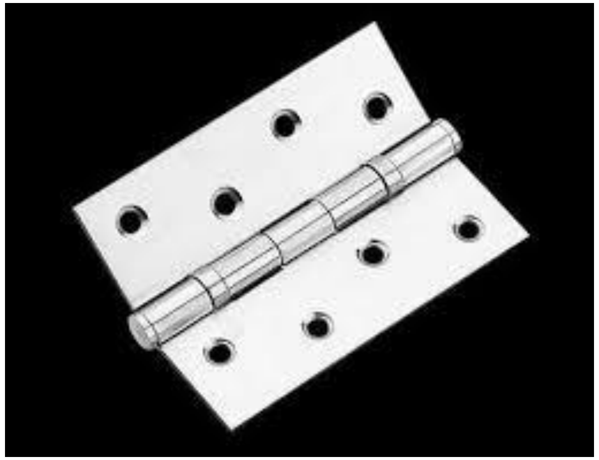
形式与带正则的线性回归或Logistic回归的目标函数类似

$$J(\boldsymbol{w},\lambda)=C\sum_{i=1}^{m}L(y_i,f(\boldsymbol{x}_i,\boldsymbol{w}))+R(\boldsymbol{w})$$

经验风险
(Empirical Risk)
描述模型与训练数据的契合程度

结构风险
(Structural Risk)
描述模型本身的某些性质

5.5 合页损失函数（Hinge Loss）

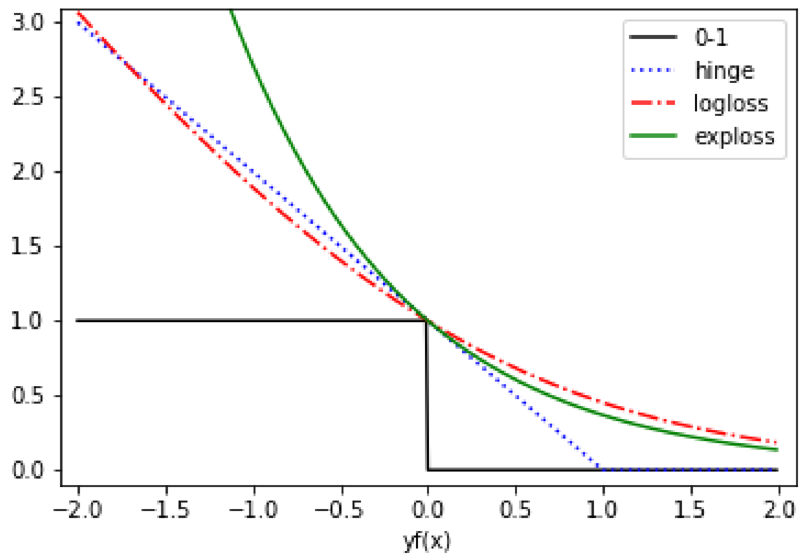


在C-SVM中

- 当$y_i\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i+b\right)\geq 1$，$\xi_i=0$
- 其他点：$\xi_i=1-y_i\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i+b\right)$

5.5.1 合页损失

$$\xi=L_{Hinge}(y,\hat{y})=\begin{cases}0 & y\hat{y}\geq 1\\ 1-y\hat{y} & otherwise\end{cases}$$



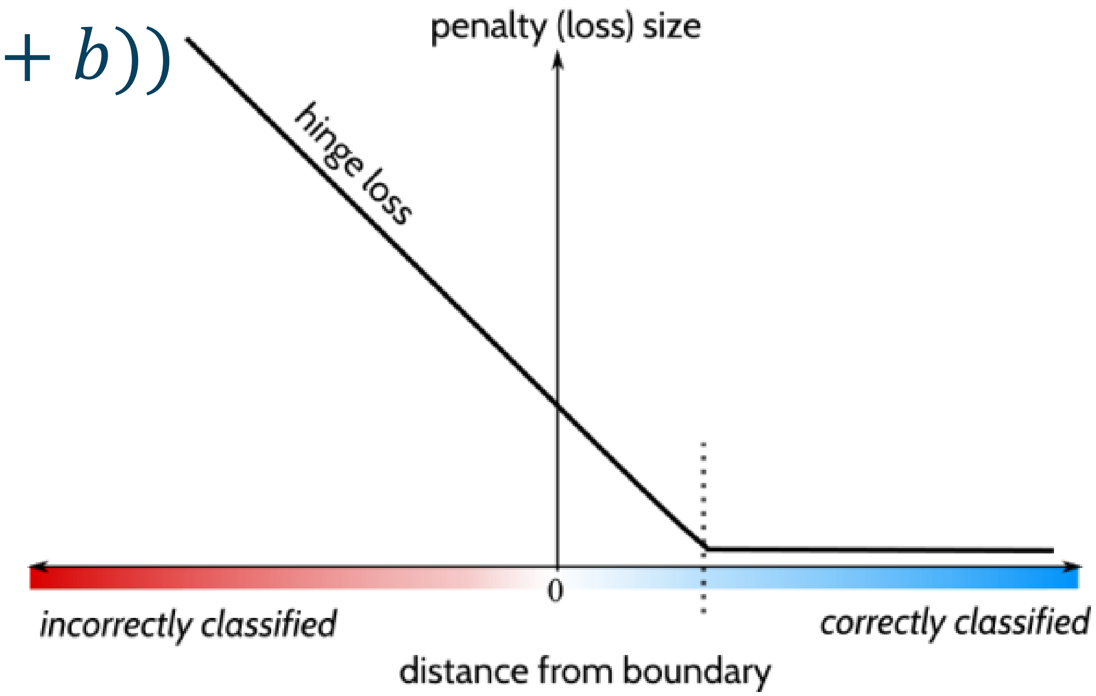
$$\xi = L_{Hinge}(y, \hat{y}) = \begin{cases} 0 & y\hat{y} \geq 1 \\ 1 - y\hat{y} & otherwise \end{cases}$$

$$\xi_i = L_{Hinge}(y_i, \hat{y}_i) = \max(0, 1 - y_i(w^T x_i + b))$$

将合页损失代入C-SVM的目标函数

$$J(\boldsymbol{w}, b, C) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{m}\xi_i$$

$$= \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{m} L_{Hinge}(y_i, \hat{y}_i)$$



合页函数 hinge loss

5.5.2 替代损失(Surrogate Loss)

软间隔SVM

替代损失函数性质较好，
一般是0/1损失函数的上界