MACHINE LEARNING

机器学习

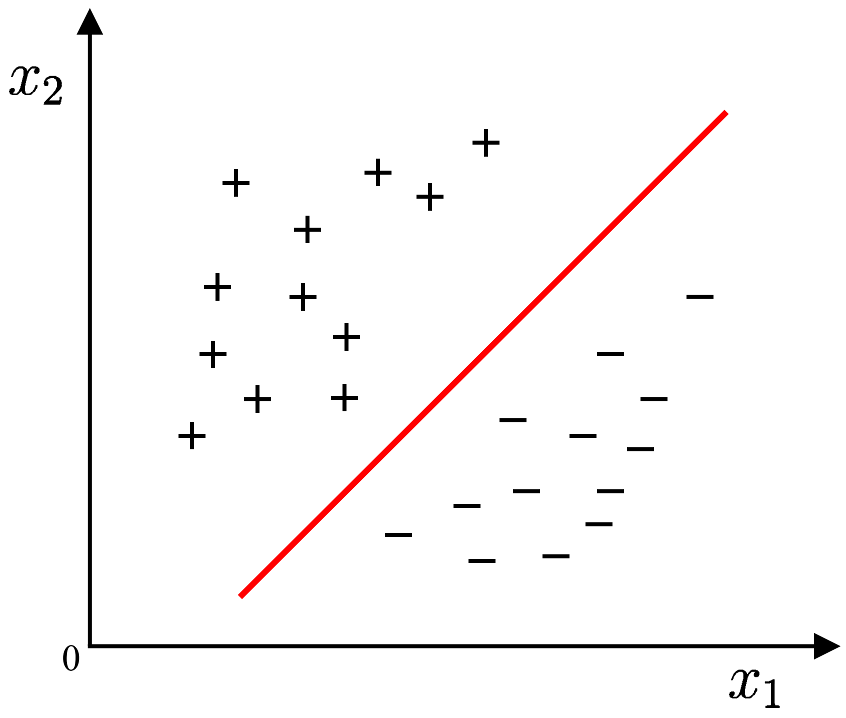
第六章：支持向量机

大纲

- 间隔与支持向量
- 对偶问题
- 核函数
- 软间隔与正则化
- 支持向量回归
- 核方法

线性模型

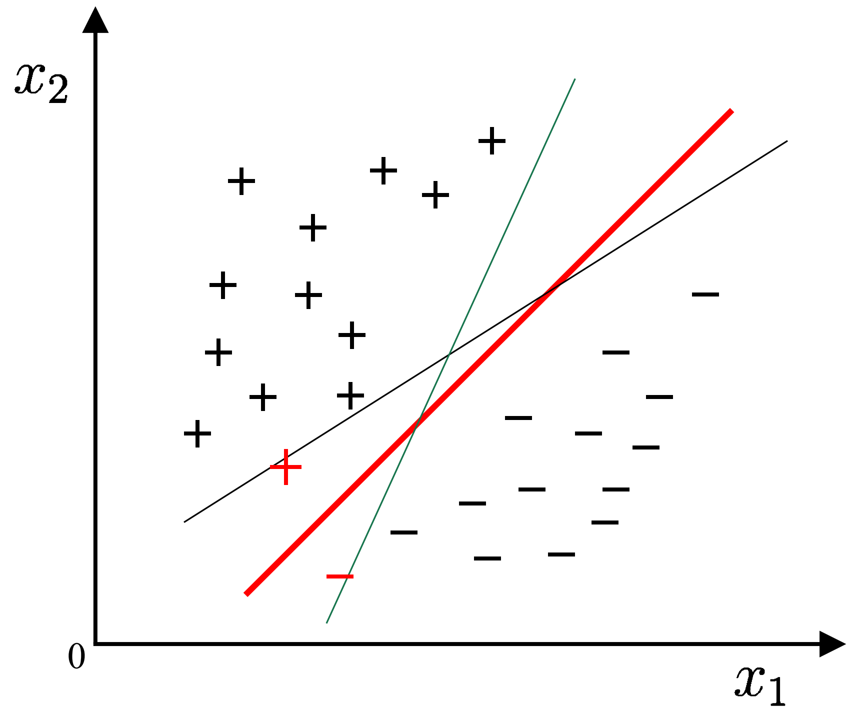
在样本空间中寻找一个超平面，将不同类别的样本分开.



引子

-Q：将训练样本分开的超平面可能有很多，哪一个好呢?

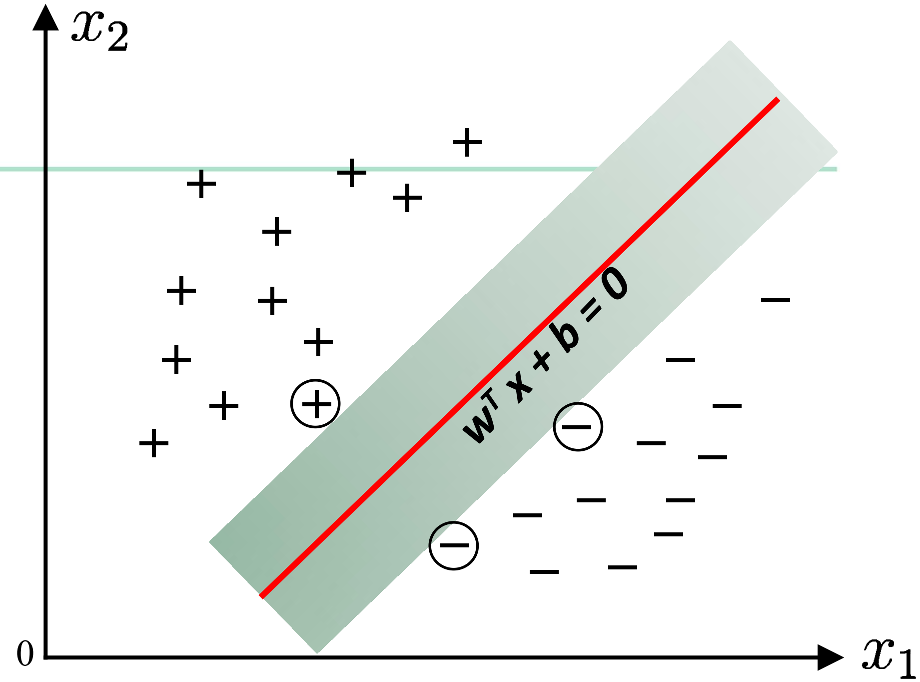
- 由于训练集的局限性或噪声的因素，训练集外的样本，可能比图中的训练样本更接近两个类的分隔界，这将使许多划分超平面出现错误
- 红色的超平面，受影响最小，所产生的分类结果，最鲁棒，对未见示例的泛化能力，最强



-A：应选择"正中间"，容忍性好，鲁棒性高，泛化能力最强的。

该划分超平面，对训练样本局部扰动的"容忍"性，最好。

间隔与支持向量

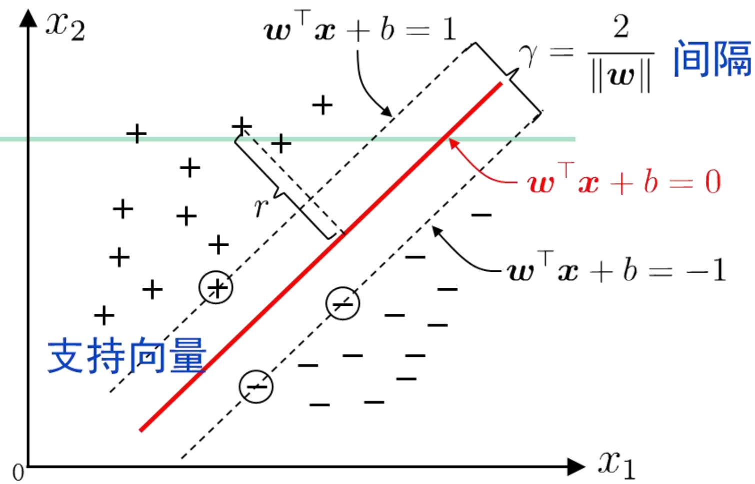


超平面方程: $\omega^T x + b = 0$

- ω 法向量，决定了超平面的方向
- b 位移项，决定了超平面与原点之间的距离

- 样本空间中任意点 x 到超平面(ω,b)的距离为 r，即 $r = \frac{|\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b|}{||\boldsymbol{w}||}$
- 假设超平面(ω,b)能将训练样本正确分类，即对于$(x_i,y_i)\in$D，
  - ✓ 若 $y_i=+1$，则有$\omega^T x_i + b > 0$；
  - ✓ 若 $y_i=-1$，则有$\omega^T x_i + b < 0$
- 距离超平面最近的这几个训练样本，称为"**支持向量**"
- 两个异类支持向量，到超平面的距离之和，称为“**间隔**”即 $\gamma = \frac{2}{||\boldsymbol{w}||}$

间隔与支持向量



- 两个异类支持向量到超平面的距离之和，

  称为“**间隔**”，$\gamma = \frac{2}{\|\boldsymbol{w}\|}$

➢ 欲找到具有“**最大间隔**”的划分超平面，即 寻找参数$\boldsymbol{\omega}$和$\boldsymbol{b}$满足约束

使得 r 最大，即

$$\arg\max_{\boldsymbol{w},b} \frac{2}{\|\boldsymbol{w}\|}$$
$$\text{s.t. } y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1,\ i = 1, 2, \ldots, m.$$

为了最大化间隔，仅需最大化$\|\boldsymbol{\omega}\|^{-1}$，等价于最小化$\|\boldsymbol{\omega}\|^{2}$，即

$$\begin{cases} w^T x_i + b \geq +1, & y_i = +1; \\ w^T x_i + b \leq -1, & y_i = -1. \end{cases}$$

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

$\{x_i \mid y_i(w^T x_i + b) = 1\}$ 称为"**支持向量**"

大纲

- 间隔与支持向量
- 对偶问题
- 核函数
- 软间隔与正则化
- 支持向量回归
- 核方法

对偶问题

- **SVM**：获得大间隔划分超平面, 即 $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{\omega}^T\boldsymbol{x}+\boldsymbol{b}$　$\boldsymbol{\omega},\boldsymbol{b}$ 是模型参数

该问题为 凸二次规划问题，即

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

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

- 使用 **拉格朗日乘子法** 可得到其"对偶问题"，即
  对上式的每条约束，添加拉格朗日乘子 $\alpha_i \geq 0$，得到

$$\max_{\alpha} \min_{w,b} L(w, b, \alpha), \quad \text{where}$$

拉格朗日函数

$$L(w,b,\alpha) = \frac{1}{2}\|w\|^2 + \sum_{i=1}^{n} \alpha_i \left(1 - y_i\left(w^T x_i + b\right)\right)$$

$$L(w,b,\alpha) = \frac{1}{2}w^T w - \sum_{i=1}^{n} \alpha_i \left(y_i\left(w^T x_i + b\right) - 1\right)$$

对偶问题

$$L(w,b,\alpha)=\frac{1}{2}w^Tw-\sum_{i=1}^{n}\alpha_i\left(y_i\left(w^Tx_i+b\right)-1\right)$$

$$\begin{cases}\dfrac{\partial}{\partial w}L(w,b,\alpha)=0\\ \dfrac{\partial}{\partial b}L(w,b,\alpha)=0\end{cases}\Rightarrow\begin{cases}w=\sum_{i=1}^{n}\alpha_i y_i x_i\\ \sum_{i=1}^{n}\alpha_i y_i=0\end{cases}$$

对偶问题

$$\begin{cases} w = \sum_{i=1}^{n} \alpha_i y_i x_i \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{cases}$$

$$L(w,b,\alpha) = \frac{1}{2} w^T w - \sum_{i=1}^{n} \alpha_i \left(y_i \left(w^T x_i + b \right) - 1 \right)$$

$$L(w,b,\alpha) = \frac{1}{2} w^T w - \sum_{i=1}^{n} \alpha_i y_i \left(w \cdot x_i \right) - b \sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i$$

0

$$\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \left(x_i \cdot x_j \right)$$

$$\sum_{i=1}^{n} \alpha_i y_i \left(x_i \cdot \sum_{j=1}^{n} \alpha_j y_j x_j \right) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \left(x_i \cdot x_j \right)$$

$$L(w,b,\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \left(x_i \cdot x_j \right)$$

对偶问题

原优化问题

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

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

可转化为 “对偶问题”，即

$$\max_{\alpha} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \left(x_i \cdot x_j\right)$$

$$s.t. \quad \alpha_i \geq 0, \quad i = 1, 2, \cdots, n$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

对偶问题

新的优化问题 $$\max_{\alpha} \quad \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j \left(x_i \cdot x_j\right)$$

$$s.t. \quad \alpha_i \geq 0, \quad i = 1, 2, \cdots, n$$

$$\sum_{i=1}^{n}\alpha_i y_i = 0$$

基于KKT条件，使用通用的二次规划算法求解，

$$w^* = \sum_{i=1}^{n}\alpha_i^* y_i x_i$$

$$b^* = y_i - w^* \bullet x^*$$

得到模型

$$f(x) = w^{*T}x + b^* = \sum_{i=1}^{m}\alpha_i^* y_i x_i^T x + b^*$$

支持向量

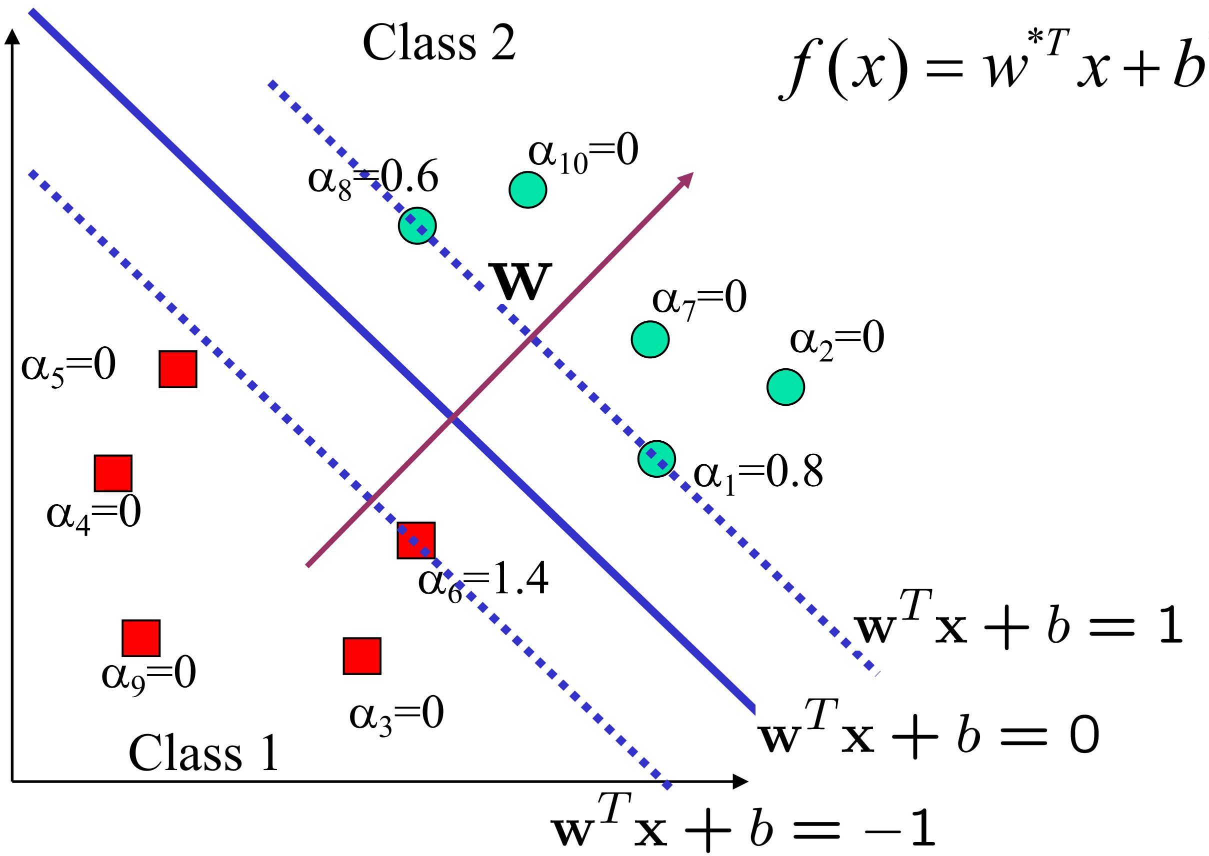
- 最终模型 $$f(x)=w^{*T}x+b^{*}=\sum_{i=1}^{m}\alpha_i^{*}y_ix_i^{T}x+b^{*}$$

- KKT条件 $$\begin{cases}\alpha_i \geq 0, \\ y_i f(\boldsymbol{x}_i) \geq 1, \\ \alpha_i(y_i f(\boldsymbol{x}_i)-1)=0.\end{cases}$$

- 于是，对于任意的训练样本 (x_i, y_i)，总有 $\alpha_i=0$ 或者 $y_i f(x_i)=1$
  - 若 $\alpha_i=0$，则该样本不会在模型中出现，也不会对 $f(x)$ 有任何影响；
  - 若 $\alpha_i>0$，则必有 $y_i f(x_i)=1$，所对应的样本点位于最大间隔的边界上

支持向量

$$f(x) = w^{*T} x + b^* = \sum_{i=1}^{m} \alpha_i^* y_i x_i^T x + b^*$$



支持向量机解的稀疏性

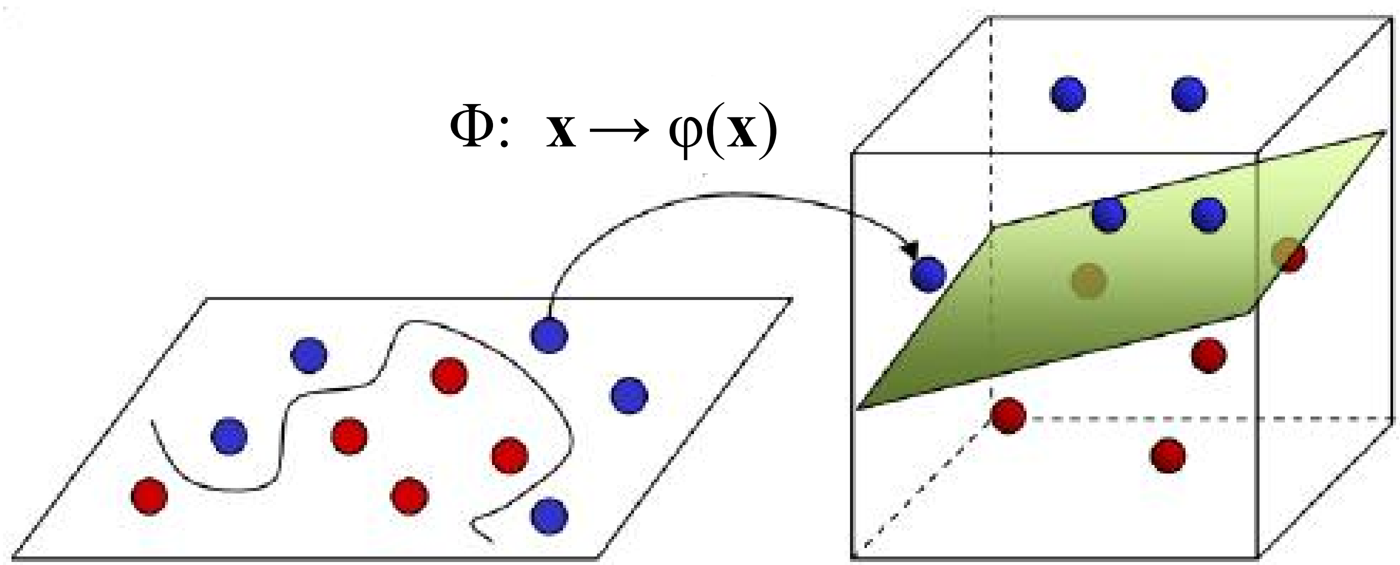
训练完成后大部分的训练样本都不需保留，最终模型仅与支持向量有关。

大纲

- 间隔与支持向量
- 对偶问题
- 核函数
- 软间隔与正则化
- 支持向量回归
- 核方法

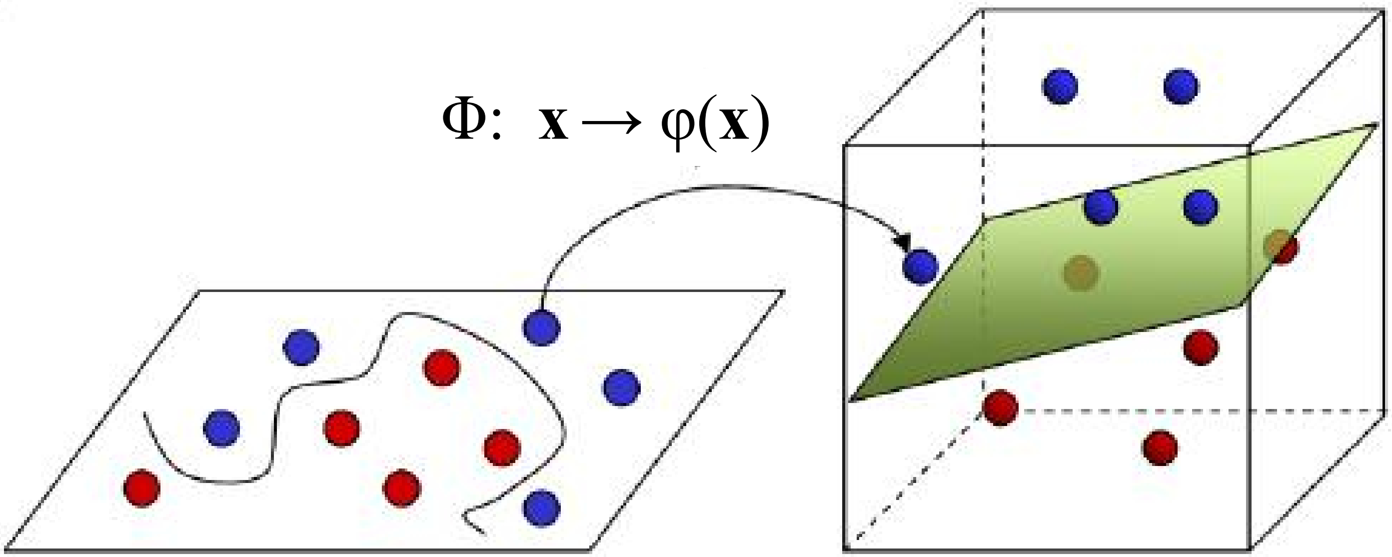
线性不可分

-**Q**: 若不存在一个能正确划分两类样本的超平面, 怎么办?



-**A**: 将样本从原始空间映射到一个更高维的特征空间, 使得样本在这个特征空间内线性可分.

线性不可分–例子



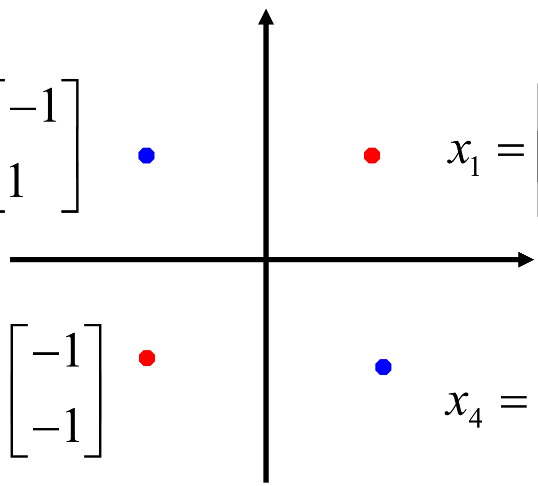
$$\phi(x): R^2 \rightarrow R^3$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Rightarrow z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} = \begin{bmatrix} x_1^{\ 2} \\ \sqrt{2}x_1x_2 \\ x_2^{\ 2} \end{bmatrix}$$

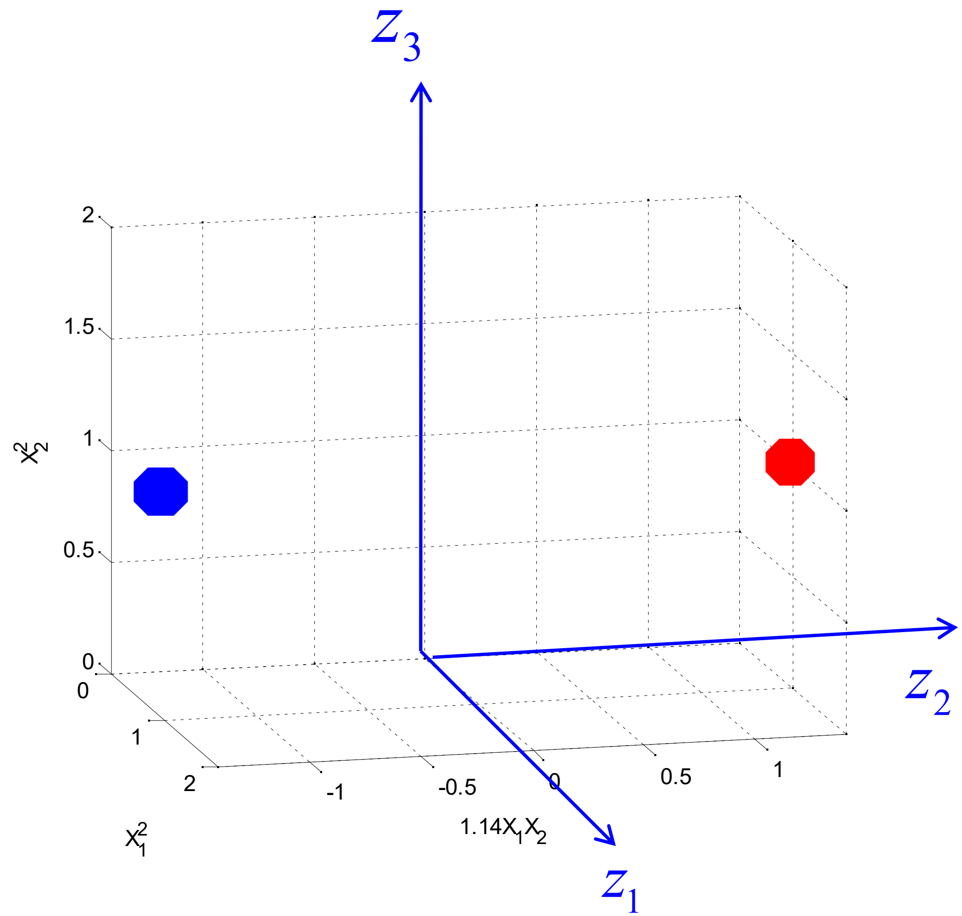
线性不可分–例子

$$\phi(x): R^2 \rightarrow R^3$$

$$x_2 = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \qquad x_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



$$x_3 = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \qquad x_4 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$



高维可分-最大间隔

- 令 $\phi(x)$ 表示将 x 映射后的特征向量，于是，在特征空间中划分超平面所对应的模型可表示为

$$f(x) = w^T \phi(x) + b$$

- 原优化问题变为 $\min\limits_{w,b} \quad \frac{1}{2}\|w\|^2$

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

- 其对偶问题是 $\max\limits_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \phi(x_i)^T \phi(x_j)$

$$s.t. \quad (1) \sum_{i=1}^{n} \alpha_i y_i = 0$$

$$(2) \ \alpha_i \geq 0, i = 1, \cdots, m.$$

挑战：高维特征空间中样本点的内积运算复杂性高

核函数

$$\phi(x): R^2 \rightarrow R^3$$

$$x_1 = \begin{bmatrix} x_{11} \\ x_{12} \end{bmatrix} \quad z_1 = \begin{bmatrix} z_{11} \\ z_{12} \\ z_{13} \end{bmatrix} = \begin{bmatrix} x_{11}{}^2 \\ \sqrt{2}x_{11}x_{12} \\ x_{12}{}^2 \end{bmatrix} \quad x_2 = \begin{bmatrix} x_{21} \\ x_{22} \end{bmatrix} \quad z_2 = \begin{bmatrix} z_{21} \\ z_{22} \\ z_{23} \end{bmatrix} = \begin{bmatrix} x_{21}{}^2 \\ \sqrt{2}x_{21}x_{22} \\ x_{22}{}^2 \end{bmatrix}$$

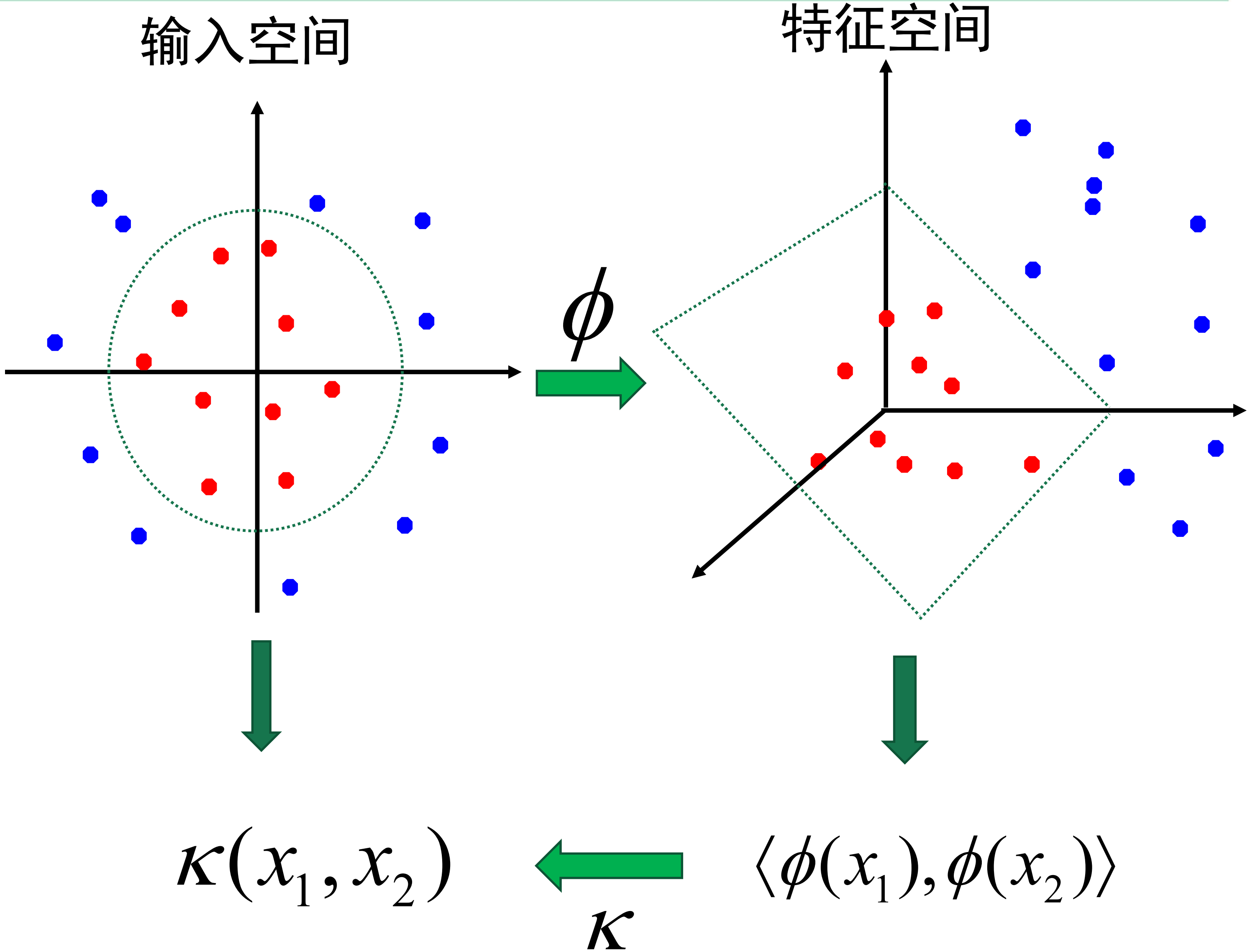
$$\phi(x_1)^T \phi(x_2) = [z_{11} \ \ z_{12} \ \ z_{13}] \begin{bmatrix} z_{21} \\ z_{22} \\ z_{23} \end{bmatrix} = z_{11}z_{21} + z_{12}z_{22} + z_{13}z_{23}$$

$$= x_{11}^2 x_{21}^2 + 2x_{11}x_{12}x_{21}x_{22} + x_{12}^2 x_{22}^2$$

$$= (x_{11}x_{21} + x_{12}x_{22})^2 = \left([x_{11} \ \ x_{12}] \begin{bmatrix} x_{21} \\ x_{22} \end{bmatrix} \right)^2$$

$$= (x_1^T x_2)^2 \equiv \kappa(x_1, x_2)$$

核函数



核函数

- 基本想法：不显式地设计核映射, 而是设计核函数.

$$\kappa(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

- Mercer定理(充分非必要)：只要一个对称函数所对应的核矩阵半正定, 则它就能作为核函数来使用.

- 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	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\\|\boldsymbol{x}_i - \boldsymbol{x}_j\\|^2}{2\delta^2}\right)$	$\delta > 0$为高斯核的带宽(width)
拉普拉斯核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\\|\boldsymbol{x}_i - \boldsymbol{x}_j\\|}{\delta}\right)$	$\delta > 0$
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, $\beta > 0$, $\theta < 0$

核支持向量机

- 设样本 x 映射后的向量为 $\phi(x)$，划分超平面为 $f(x)=\omega^T\phi(x)+b$.

原始问题

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

对偶问题

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

预测

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

大纲

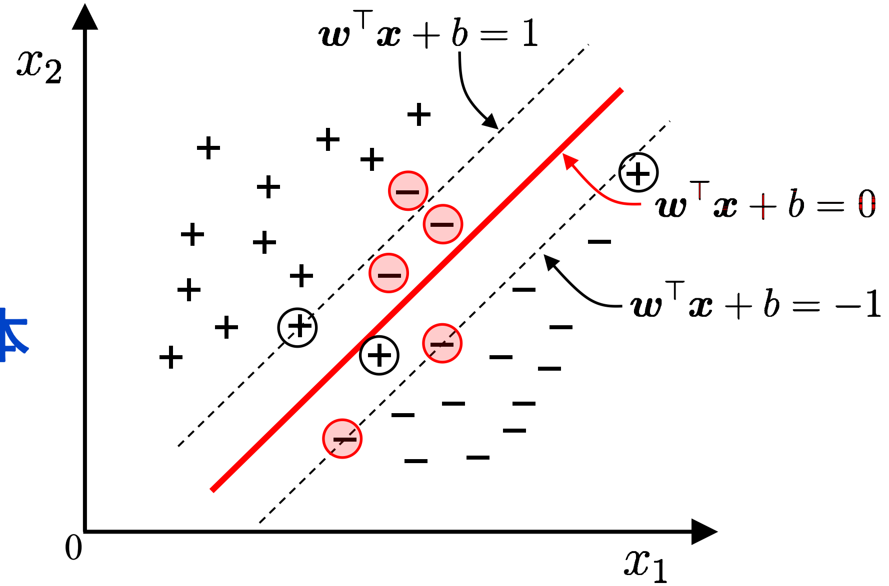
- 间隔与支持向量
- 对偶问题
- 核函数
- 软间隔与正则化
- 支持向量回归
- 核方法

软间隔

-Q:

- 在前面的讨论中，我们直假定训练样本在样本空间或特征空间中是线性可分的，即存在一个超平面能将不同类的样本完全划分开
- 现实中，很难确定合适的核函数使得训练样本在特征空间中线性可分；同时一个线性可分的结果也很难断定是否是有过拟合造成的.

不满足约束的样本



-A: 解决方法：引入”软间隔”的概念，允许支持向量机在一些样本上不满足约束.

软间隔

- 软间隔基本想法：最大化间隔的同时, 让不满足约束的样本应尽可能少.

$$\min_{\boldsymbol{w},b} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m} l_{0/1}\left(y_i(\boldsymbol{w}^\top\phi(\boldsymbol{x}_i)+b)-1\right)$$

- 其中，$l_{0/1}$ 是”0/1损失函数”

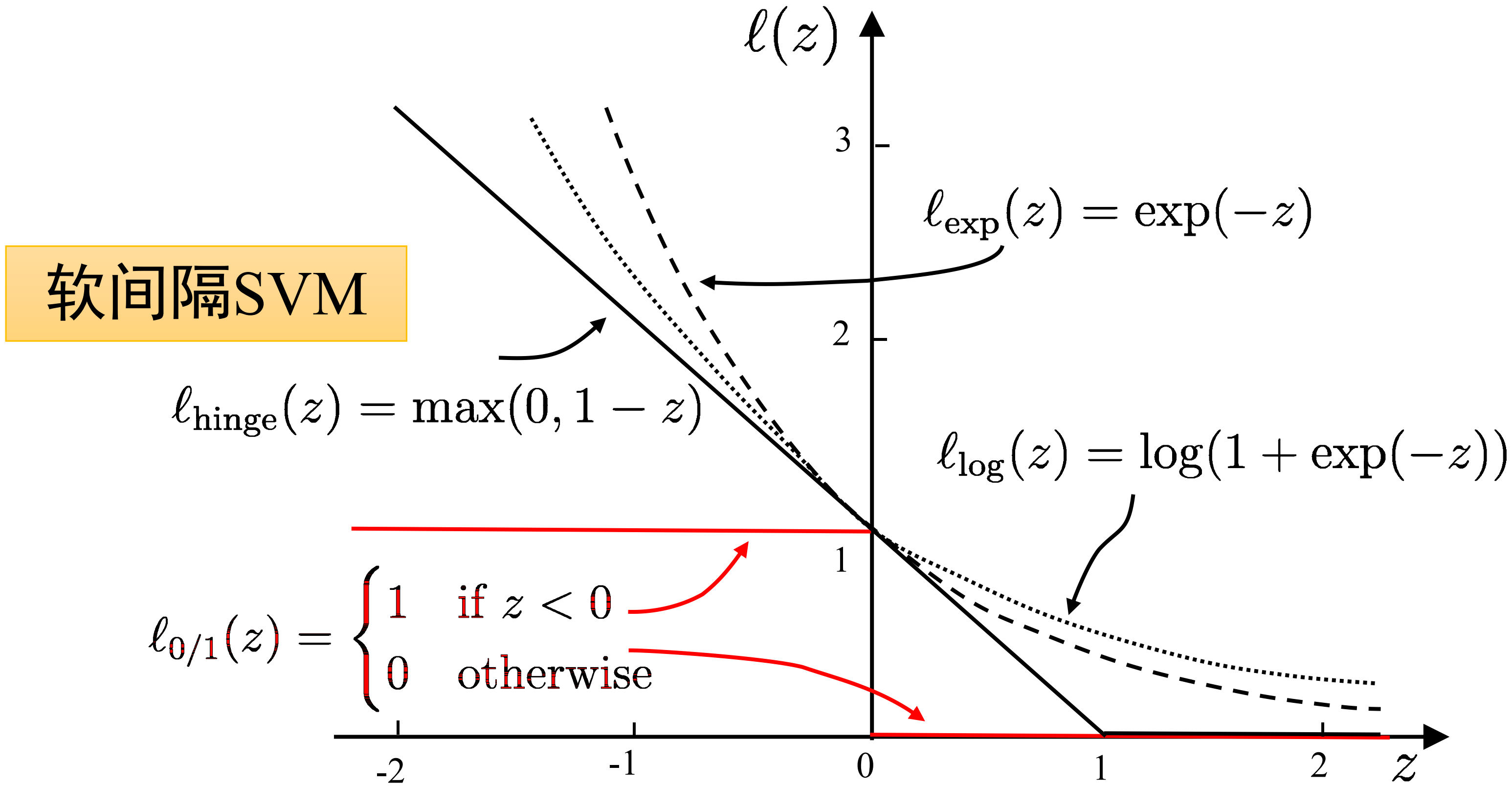
$$l_{0/1} = \begin{cases} 1 & z<0 \\ 0 & otherwise \end{cases}$$

- 存在的问题：0/1损失函数非凸、非连续, 不易优化！

- 解决方式：常用其他一些函数来代替0/1损失函数，称为“替代损失”。

  例如 hinge 损失，指数损失，对率损失

替代损失—常见的3种



替代损失函数数学性质较好，通常是凸的、连续函数
一般是0/1损失函数的上界

替代损失

hinge 损失

$$\ell_{\text{hinge}}(z) = \max(0, 1 - z)$$

原始问题

$$\min_{\boldsymbol{w},b} \ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m} \max\left(0, 1 - y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b)\right)$$

$\Downarrow$ 拉格朗日乘子法

对偶问题

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i\alpha_j y_i y_j \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j) - \sum_{i=1}^{m}\alpha_i$$

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

根据KKT条件可推得：最终模型仅与支持向量有关，也即hinge损失函数依然保持了支持向量机解的稀疏性.

正则化

- 支持向量机学习模型的更一般形式

$$\min_f \ \Omega(f) + C\sum_{i=1}^{m} l(f(\boldsymbol{x}_i), y_i)$$

结构风险，描述模型的某些性质
如，划分超平面的"间隔"大小

经验风险，描述模型与训练数据的契合程度。如，训练集上的误差

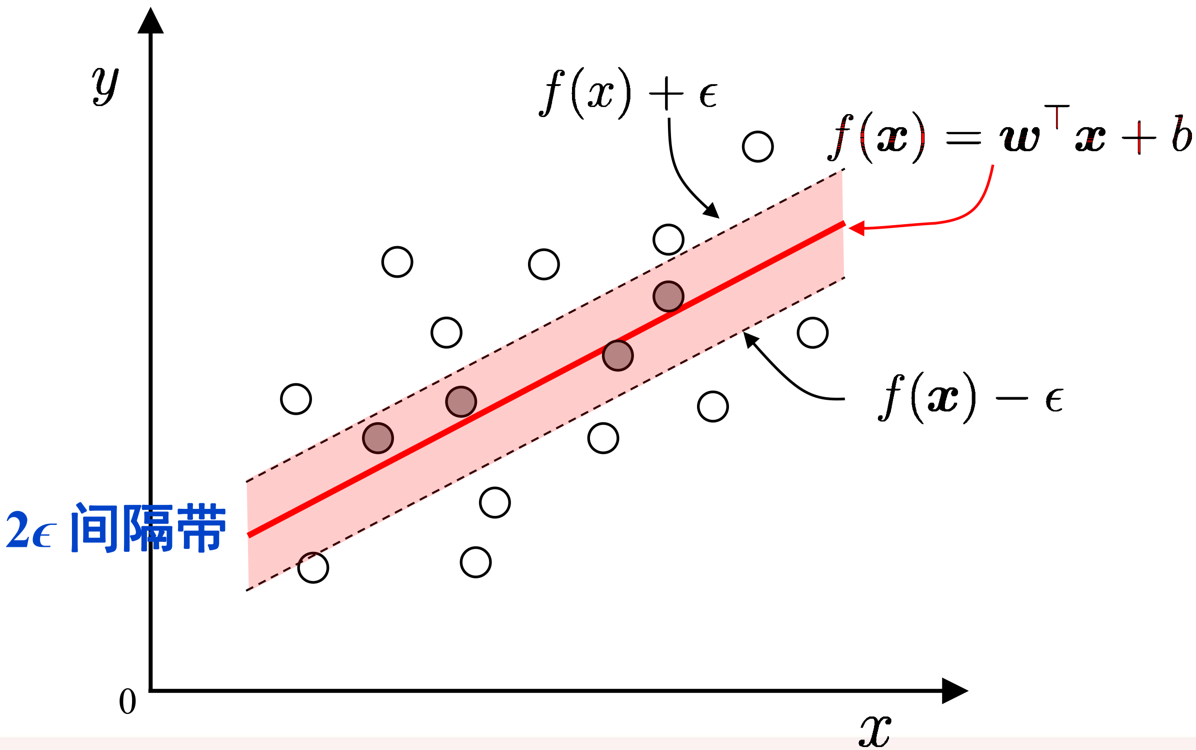
- 通过替换上面两个部分，可以得到许多其他学习模型
  - 对数几率回归(Logistic Regression)
  - 最小绝对收缩选择算子(LASSO)
  - ……

大纲

- 间隔与支持向量
- 对偶问题
- 核函数
- 软间隔与正则化
- 支持向量回归
- 核方法

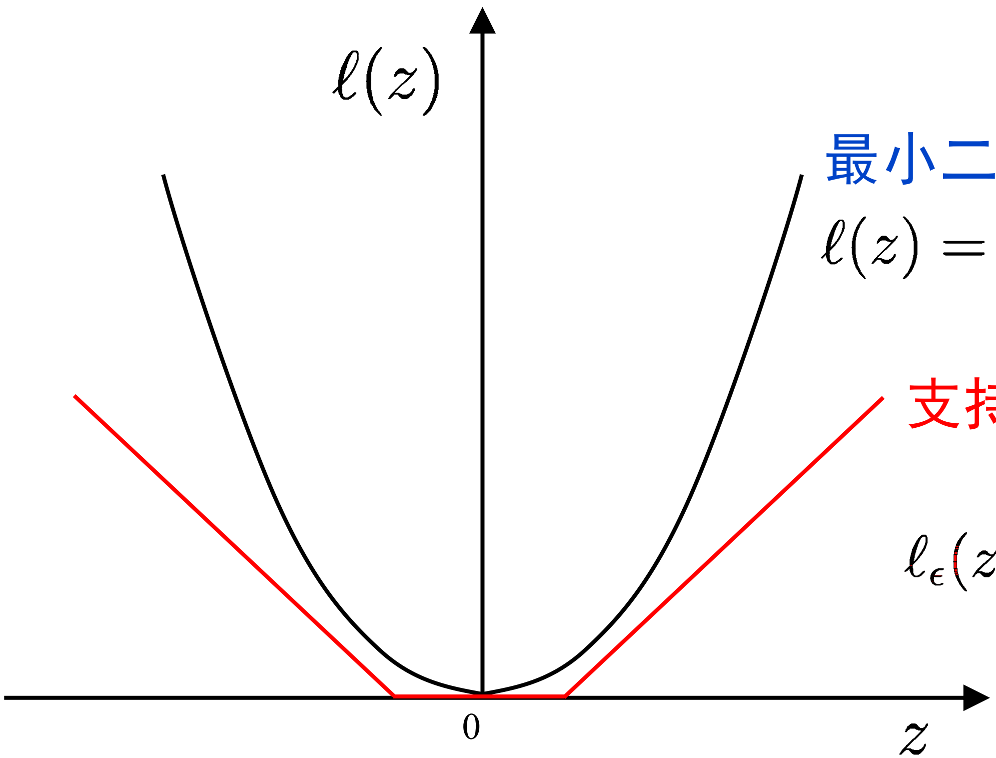
支持向量回归 SVR

- 传统回归模型通常直接基于模型输出f(x)与真实输出y之间的差别来计算损失，当且仅当 f(x) 与 y 完全相同时，损失才为零.
- 支持向量回归特点：允许模型输出和实际输出间存在 2ϵ 的偏差.



支持向量回归 SVR --损失函数

落入中间 2ϵ 间隔带的样本不计算损失，从而使得模型获得稀疏性.



最小二乘损失函数

$$\ell(z) = z^2$$

支持向量回归损失函数

$$\ell_\epsilon(z) = \begin{cases} 0 & \text{if } |z| \le \epsilon \\ |z| - \epsilon & \text{otherwise} \end{cases}$$

支持向量回归 SVR --形式化

原始问题

$$\min_{\boldsymbol{w},b,\xi_i,\hat{\xi}_i} \quad \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}(\xi_i + \hat{\xi}_i)$$

拉格朗日乘子法

对偶问题

$$\min_{\boldsymbol{\alpha},\hat{\boldsymbol{\alpha}}} \quad \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i - \hat{\alpha}_i)(\alpha_j - \hat{\alpha}_j)\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) + \sum_{i=1}^{m}(\alpha_i(\epsilon - y_i) + \hat{\alpha}_i(\epsilon + y_i))$$

预测

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top\phi(\boldsymbol{x}) + b = \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i)y_i\kappa(\boldsymbol{x}_i, \boldsymbol{x}) + b$$

大纲

- 间隔与支持向量
- 对偶问题
- 核函数
- 软间隔与正则化
- 支持向量回归
- 核方法

表示定理

支持向量机 SVM $f(\boldsymbol{x}) = \boldsymbol{w}^\top \phi(\boldsymbol{x}) + b = \sum_{i=1}^{m} \alpha_i y_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}) + b$

支持向量回归 SVR $f(\boldsymbol{x}) = \boldsymbol{w}^\top \phi(\boldsymbol{x}) + b = \sum_{i=1}^{m} (\hat{\alpha}_i - \alpha_i) y_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}) + b$

结论：无论是支持向量机还是支持向量回归，若不考虑偏移项b，学得的模型总可以表示成核函数的线性组合.

更一般的结论(表示定理): 对于任意单调增函数Ω 和任意非负损失函数l，优化问题 $\min_{h \in \mathbb{H}} \quad F(h) = \Omega(\|h\|_{\mathbb{H}}) + l(h(\boldsymbol{x}_1), \ldots, h(\boldsymbol{x}_m))$

的解总可以写为 $h^* = \sum_{i=1}^{m} \alpha_i \kappa(\cdot, \boldsymbol{x}_i)$

- 对损失函数没有限制，对正则化项Ω仅要求单调递增，甚至不要求Ω是凸函数，意味着对于一般的损失函数和正则化项，优化问题的最优解h*(x)都可表示为核函数κ 的线性组合，这显示出核函数的巨大威力.

核线性判别分析

- 通过表示定理可以得到很多线性模型的"核化"版本
  - 核SVM、核LDA、核PCA、……
  - 通过"核化" (即引入核函数)来将线性学习器拓展为非线性学习器.

总结

- 支持向量机的"最大间隔"思想
- 对偶问题及其解的稀疏性
- 通过向高维空间映射解决线性不可分的问题
- 引入"软间隔"缓解特征空间中线性不可分的问题
- 将支持向量的思想应用到回归问题上得到支持向量回归
- 将核方法推广到其他学习模型