第七章

支持向量机

袁春 清华大学深圳研究生院
李航 华为诺亚方舟实验室

目录

一、线性可分支持向量机与硬间隔最大化

- 线性可分支持向量机
- 函数间隔和几何间隔
- 间隔最大化
- 学习的对偶算法

线性可分支持向量机

- 二分类问题：
- 输入空间：欧式空间或离散集合
- 特征空间：欧式空间或希尔伯特空间
- 线性可分支持向量机、线性支持向量机：假设这两个空间的元素一一对应，并将输入空间中的输入映射为特征空间中的特征向量；
- 非线性支持向量机：利用一个从输入空间到特征空间的非线性映射将输入映射为特征向量；
- 支持向量机的学习是在特征空间进行的.

线性可分支持向量机

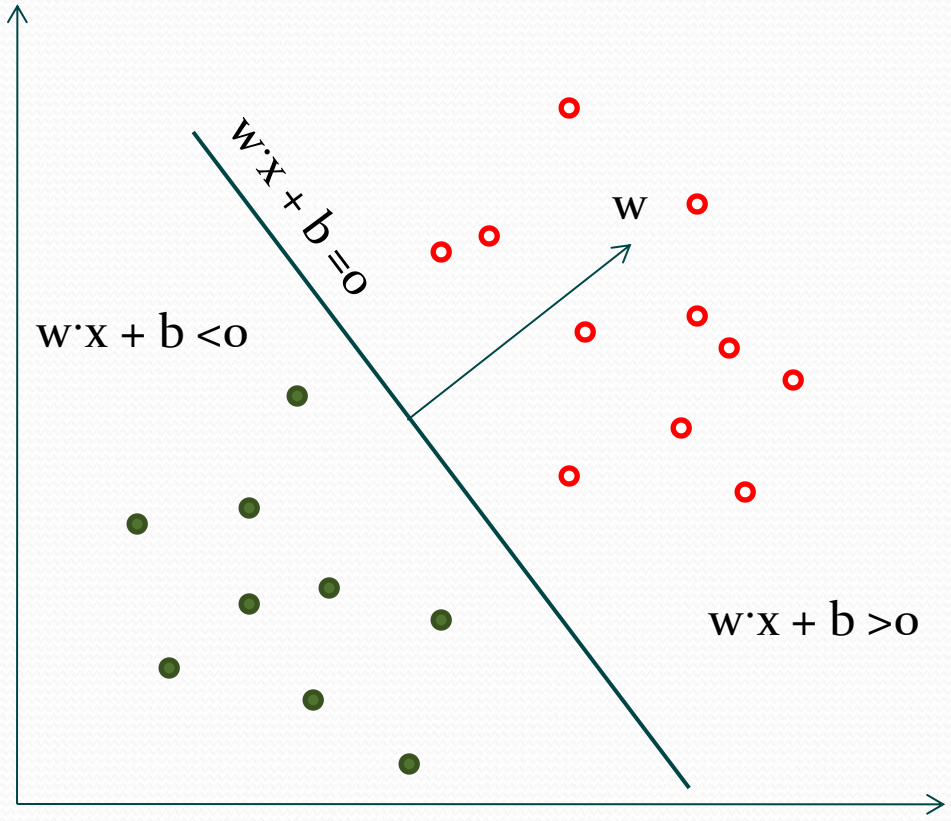
- 假设特征空间上的训练数据集：

$$T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$$

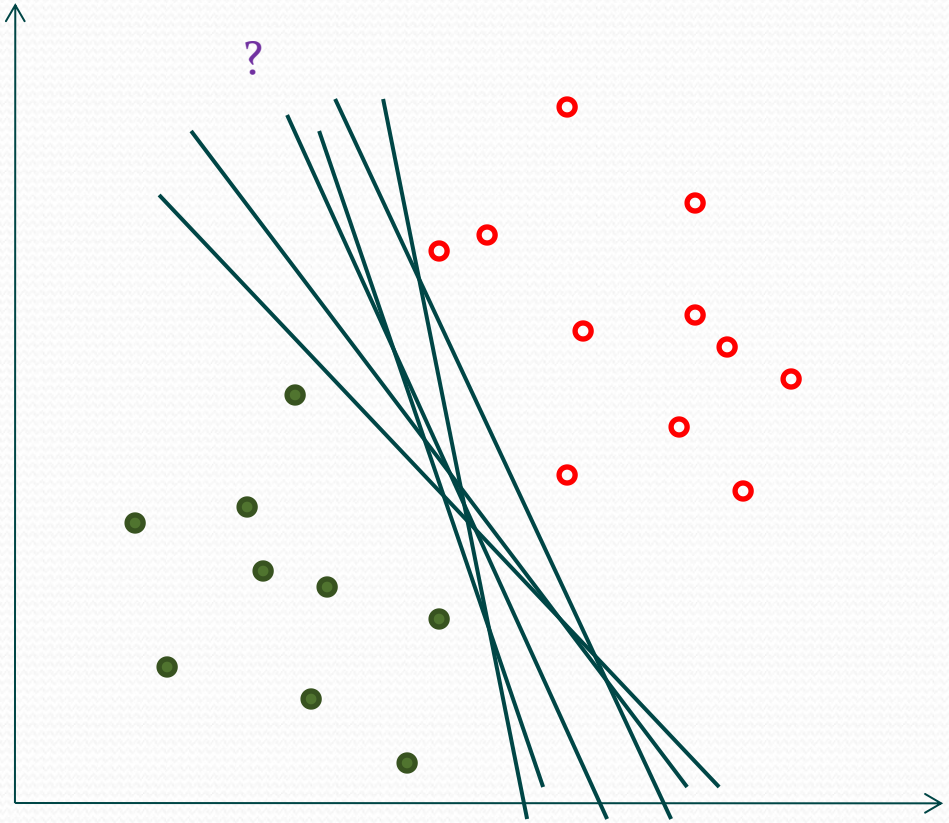
$$x_i\in\mathcal{X}=\mathbf{R}^n,\quad y_i\in\mathcal{Y}=\{+1,-1\},\quad i=1,2,\cdots,N$$

- 正例和负例
- 学习的目标：找到分类超平面，
- 线性可分支持向量机：给定线性可分训练数据集，通过间隔最大化或等价地求解相应的凸二次规划问题学习得到的分离超平面为 $w^*\cdot x+b^*=0$
- 决策函数：$f(x)=\text{sign}(w^*\cdot x+b^*)$

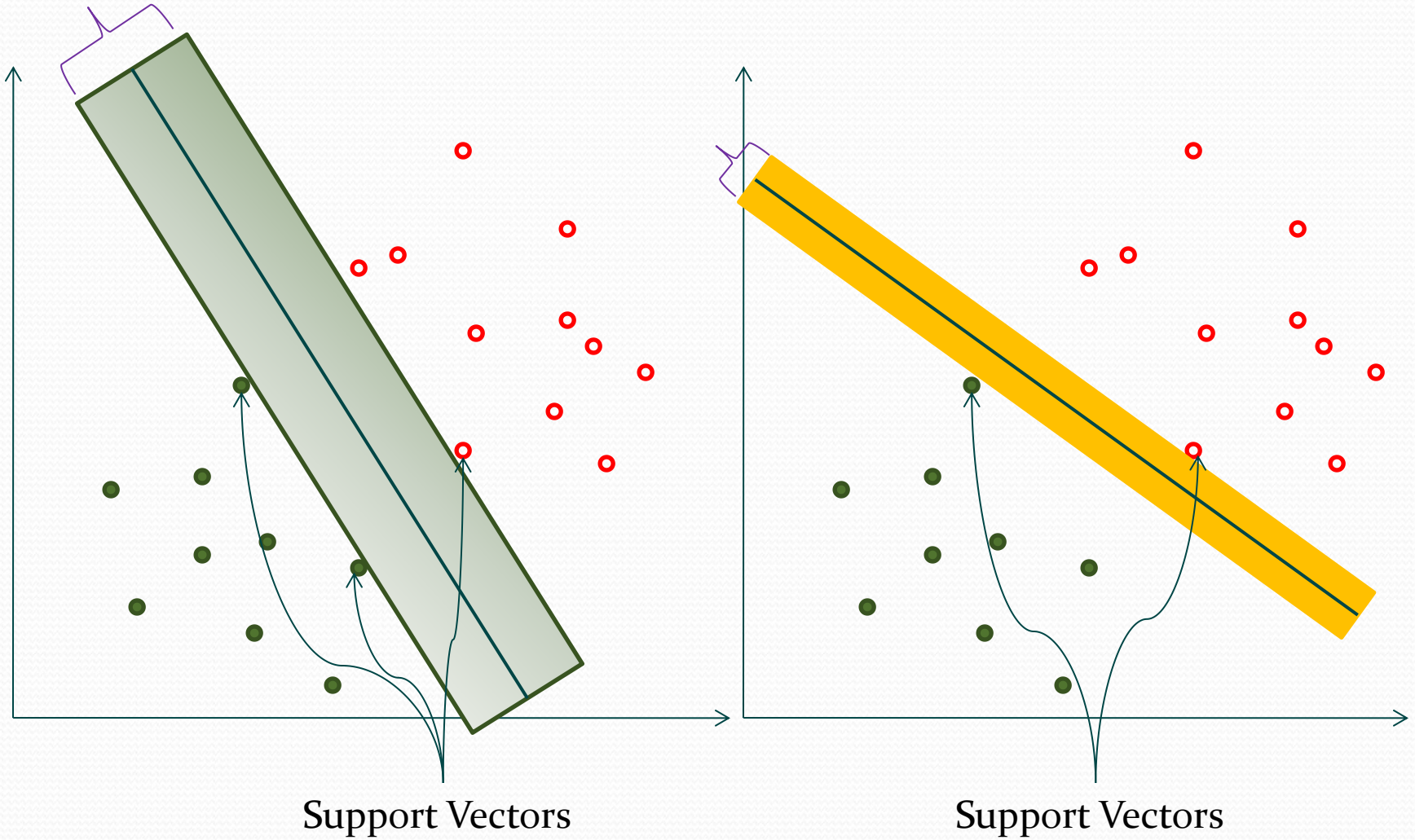
线性可分支持向量机与硬间隔最大化



超平面选择

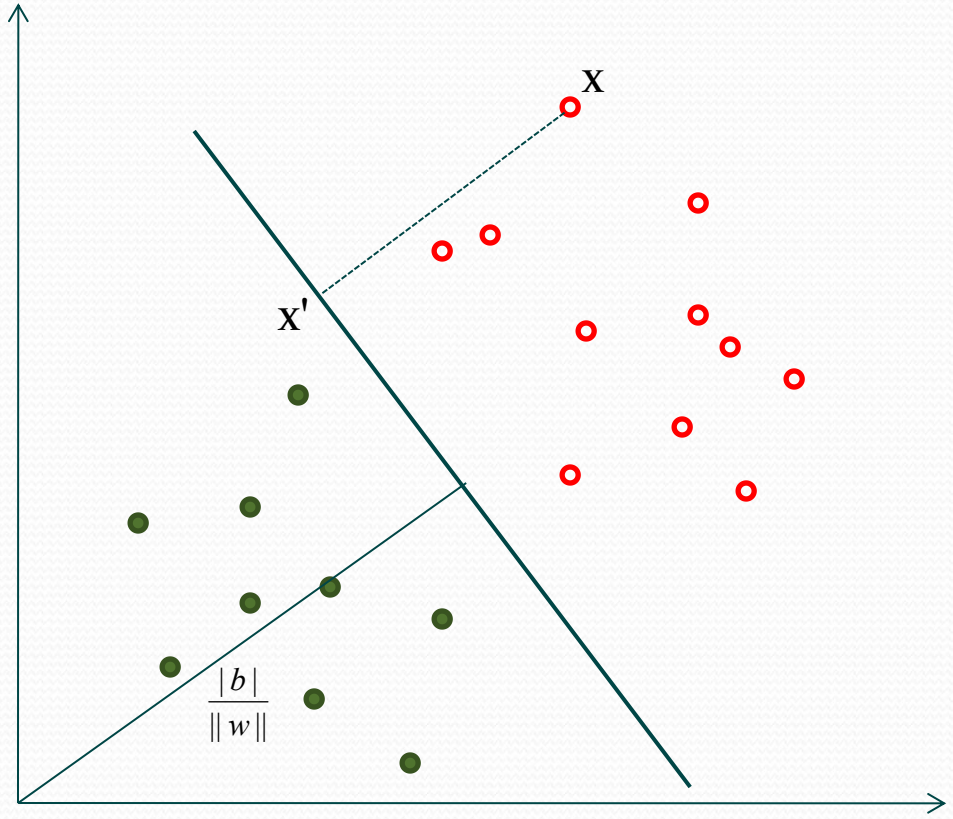


Margins



点到超平面的距离

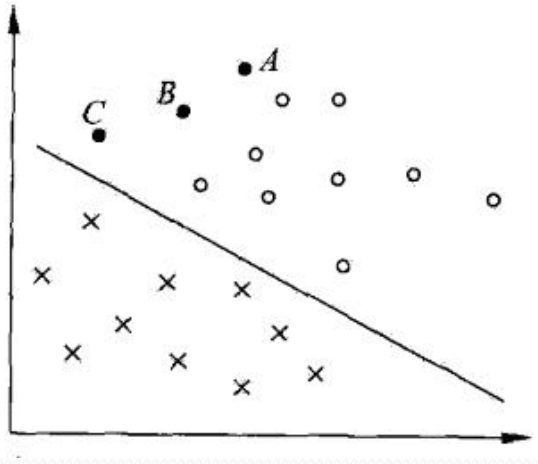
$$g(x) = w \cdot x + b$$



函数间隔和几何间隔

- 点到分离超平面的远近 $|w \cdot x + b|$
- ➡ 表示分类预测的确信程度
- $w \cdot x + b$ 的符号与类标记y的符号是否一致
- ➡ 表示分类是否正确

- 所以：
- $y(w \cdot x + b)$ 表示分类的正确性和确信度



函数间隔和几何间隔

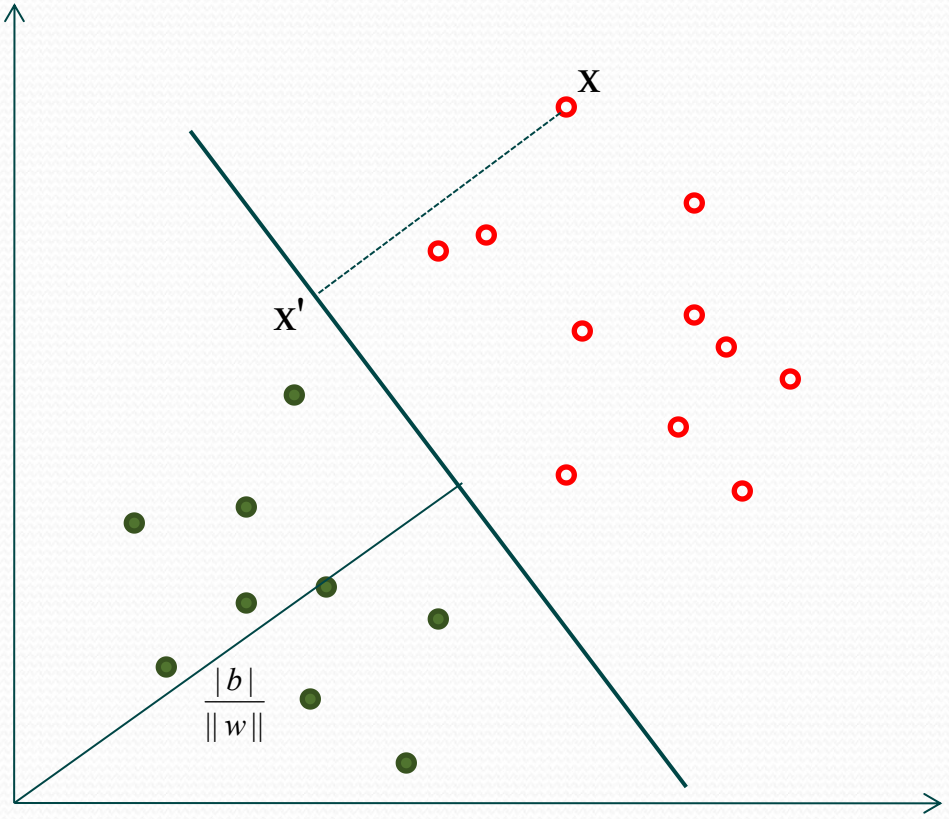
- 函数间隔
  - 样本点的函数间隔

$$\hat{\gamma}_i = y_i(w \cdot x_i + b)$$

  - 训练数据集的函数间隔

$$\hat{\gamma} = \min_{i=1,\cdots,N} \hat{\gamma}_i$$

  - 表示分类预测的正确性和确信度
  - 当成比例改变w和b

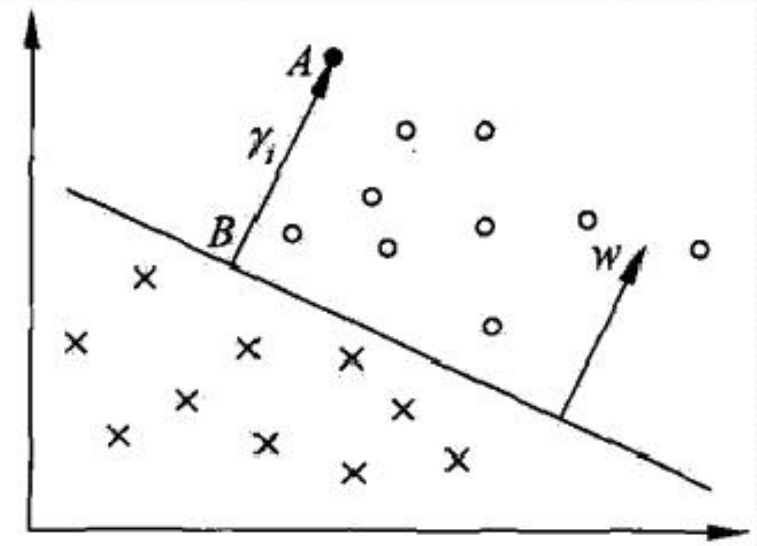


函数间隔和几何间隔

- 几何间隔
  - 样本点的几何间隔：正例和负例

$$\gamma_i = y_i\left(\frac{w}{\|w\|}\cdot x_i + \frac{b}{\|w\|}\right) \qquad \gamma_i = -\left(\frac{w}{\|w\|}\cdot x_i + \frac{b}{\|w\|}\right)$$

$$\gamma_i = y_i\left(\frac{w}{\|w\|}\cdot x_i + \frac{b}{\|w\|}\right)$$



函数间隔和几何间隔

- 几何间隔
  - 对于给定的训练数据集T和超平面(w, b)

$$\gamma_i = y_i\left(\frac{w}{\|w\|}\cdot x_i + \frac{b}{\|w\|}\right)$$

  - 训练数据集的几何间隔

$$\gamma = \min_{i=1,\cdots,N}\gamma_i$$

  - 即

$$\gamma_i = \frac{\hat{\gamma}_i}{\|w\|}$$

$$\gamma = \frac{\hat{\gamma}}{\|w\|}$$

间隔最大化

- 最大间隔分类超平面

$$\max_{w,b} \quad \gamma$$

$$\text{s.t.} \quad y_i\left(\frac{w}{\|w\|}\cdot x_i+\frac{b}{\|w\|}\right)\geqslant\gamma, \quad i=1,2,\cdots,N$$

- 根据几何间隔和函数间隔的关系

$$\max_{w,b} \quad \frac{\hat{\gamma}}{\|w\|}$$

$$\text{s.t.} \quad y_i(w\cdot x_i+b)\geqslant\hat{\gamma}, \quad i=1,2,\cdots,N$$

- 考虑
  - 可以取 $\hat{\gamma}=1$
  - 最大化 $\frac{1}{\|w\|}$ 和最小化 $\frac{1}{2}\|w\|^2$ 等价

间隔最大化

- 线性可分支持向量机学习的最优化问题

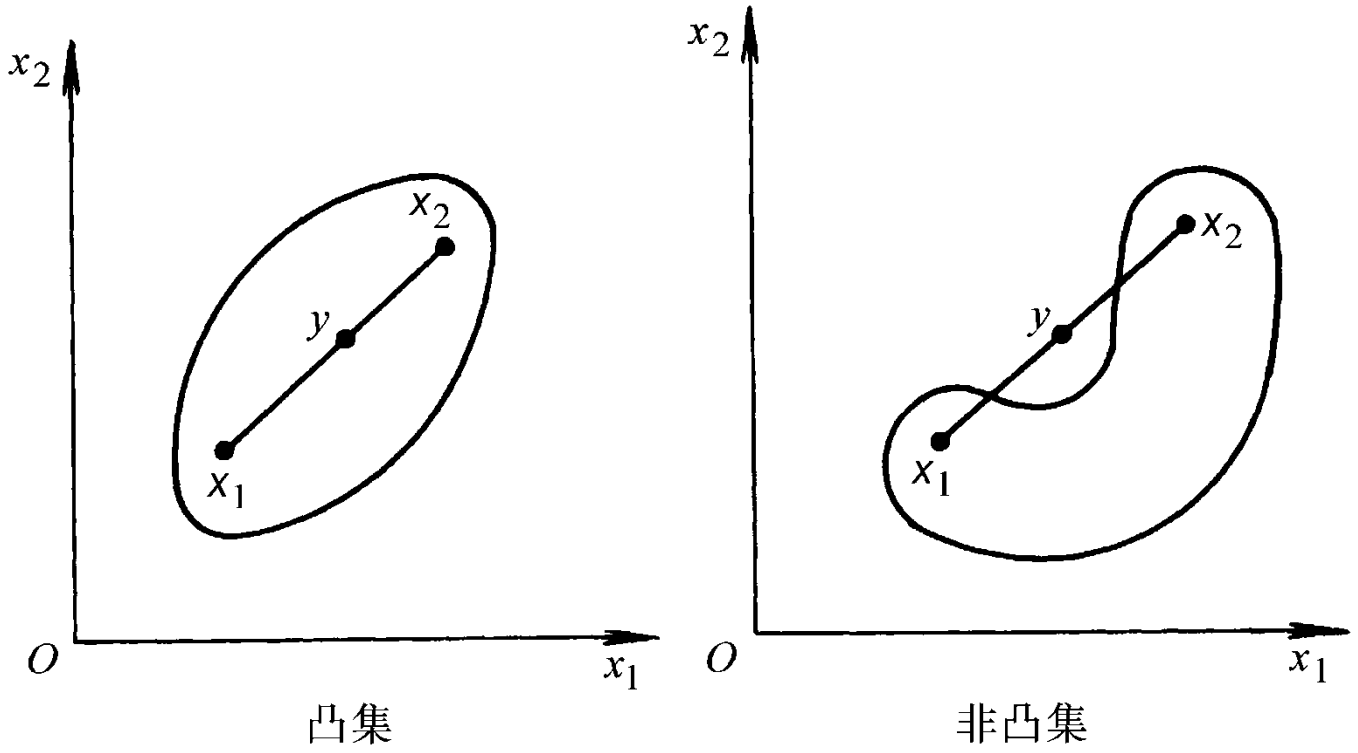
$$\min_{w,b} \quad \frac{1}{2}\|w\|^2$$

$$\text{s.t.} \quad y_i(w \cdot x_i + b) - 1 \geqslant 0, \quad i = 1, 2, \cdots, N$$

- 凸二次规划(convex quadratic programming)

凸集

一个点集（或区域），如果连接其中任意两点 x_1 x_2 的线段都全部包含在该集合内，就称该点集为凸集，否则为非凸集。



凸性条件

1.根据一阶导数（函数的梯度）来判断函数的凸性

设f(x)为定义在凸集R上，且具有连续的一阶导数的函数，则f(x)在R上为凸函数的充要条件是对凸集R内任意不同两点x_1 x_2，不等式

$$f(x_2) \geq f(x_1) + (x_2 - x_1)^T \nabla f(x_1)$$

恒成立。

凸性条件

2.根据二阶导数（ Hesse矩阵)来判断函数的凸性

设f(x)为定义在凸集R上且具有连续二阶导数的函数，则f(x)在R上为凸函数的充要条件：

Hesse矩阵在R上处处半正定

凸规划

对于约束优化问题

$$\min f(x)$$

$$s.t. \quad g_j(x) \le 0 \quad j = 1, 2, ..., m$$

若 $f(x)$ $g_j(x)$都为凸函数，则此问题为凸规划。

凸规划的性质

1.若给定一点x^0，则集合 $R=\left\{x \middle| f(x)\le f(x_0)\right\}$ 为凸集。

2.可行域 $R=\left\{x \middle| g_j(x)\le 0 \quad j=1,2,\ldots,m\right\}$ 为凸集

3.凸规划的任何局部最优解就是全局最优解

凸优化问题

- 凸优化问题: 指约束最优化问题

$$\min_{w} \quad f(w)$$
$$\text{s.t.} \quad g_i(w) \leqslant 0, \quad i = 1,2,\cdots,k$$
$$h_i(w) = 0, \quad i = 1,2,\cdots,l$$

- 其中，目标函数f(w) 和约束函数$g_i(w)$都是R^n上连续可微的凸函数，约束函数$h_j(w)$是R^n上的仿射函数
- 当目标函数为二次函数，g函数为仿射函数时，为凸二次规划问题。

线性可分支持向量机学习算法

- 输入：线性可分训练数据集 $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$

$$x_i \in \mathcal{X}=\mathbf{R}^n \qquad y_i \in \mathcal{Y}=\{-1,+1\},\quad i=1,2,\cdots,N$$

- 输出：最大间隔分离超平面和分类决策函数
- 1、构造并求解约束最优化问题 (7.13) ～ (7.14) 1

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2$$
$$\text{s.t.} \quad y_i(w\cdot x_i+b)-1\geqslant 0,\quad i=1,2,\cdots,N$$

求得w*和b*

- 2、得到分离超平面

$$w^*\cdot x+b^*=0$$

- 分类决策函数

$$f(x)=\text{sign}(w^*\cdot x+b^*)$$

支持向量和Margins（边界）

- 在线性可分情况下，训练数据集的样本点中与分离超平面距离最近的样本点的实例称为支持向量(support vector)；
- 支持向量是使约束条件式等号成立的点，即

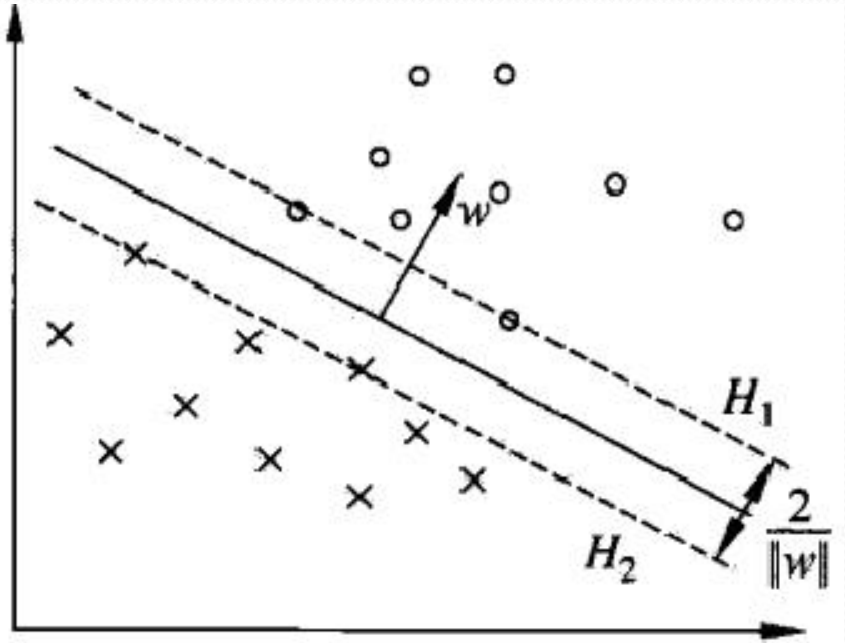
$$y_i(w \cdot x_i + b) - 1 = 0$$

- 正例：

$$H_1 : w \cdot x + b = 1$$

- 负例：

$$H_2 : w \cdot x + b = -1$$



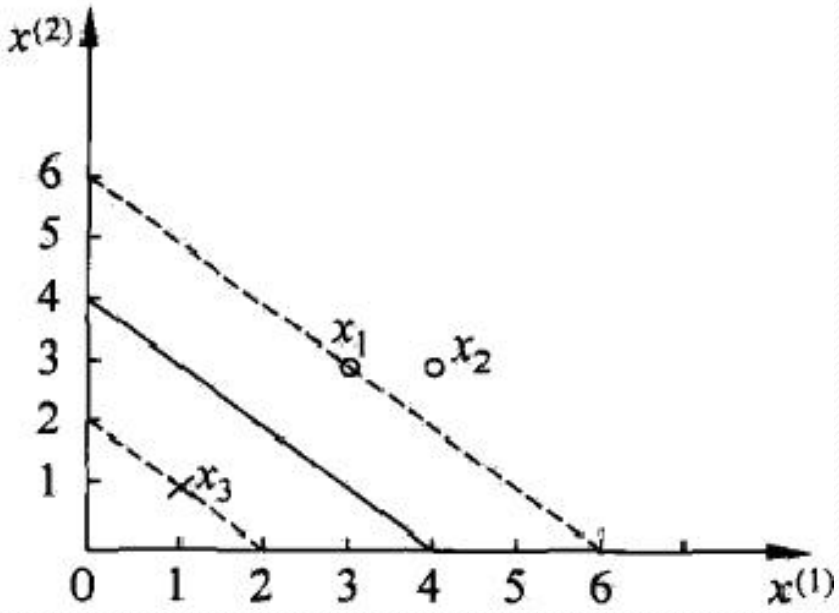
例：

$$\min_{w,b} \quad \frac{1}{2}(w_1^2 + w_2^2)$$

$$\text{s.t.} \quad 3w_1 + 3w_2 + b \geqslant 1$$

$$4w_1 + 3w_2 + b \geqslant 1$$

$$-w_1 - w_2 - b \geqslant 1$$



$$w_1 = w_2 = \frac{1}{2}, \quad b = -2$$

$$\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2 = 0$$

$x_1 = (3,3)^T$ 与 $x_3 = (1,1)^T$ 为支持向量

拉格朗日对偶

- 如何求解：(7.13) ~ (7.14)
- 在约束最优化问题中，常常利用拉格朗日对偶性(Lagrange duality)将原始问题转换为对偶问题，通过解对偶问题得到原始问题的解

1

拉格朗日对偶

- 1、原始问题
  - 设f(x),c(x),h(x)是定义在R^n上的连续可微函数

$$\min_{x\in R^n} f(x)$$

$$\text{s.t.} \quad c_i(x)\leqslant 0, \quad i=1,2,\cdots,k$$

$$h_j(x)=0, \quad j=1,2,\cdots,l$$

  - 引进拉格朗日函数 α_i, β_j 为乘子 $\alpha_i \geqslant 0$

$$L(x,\alpha,\beta)=f(x)+\sum_{i=1}^{k}\alpha_i c_i(x)+\sum_{i=1}^{l}\beta_j h_j(x)$$

  - 考虑x的函数，P为原始问题

$$\theta_P(x)=\max_{\alpha,\beta:\alpha_i\geqslant 0} L(x,\alpha,\beta)$$

拉格朗日对偶

$$\min_{x \in \mathbf{R}^n} \ f(x)$$

$$\text{s.t.} \quad c_i(x) \leqslant 0, \quad i = 1, 2, \cdots, k$$

$$h_j(x) = 0, \quad j = 1, 2, \cdots, l$$

- α_i, β_j 为乘子 $\alpha_i \geqslant 0$

$$L(x, \alpha, \beta) = f(x) + \sum_{i=1}^{k} \alpha_i c_i(x) + \sum_{j=1}^{l} \beta_j h_j(x)$$

- 假设给定某个x，如果x违反约束条件：

$$c_i(w) > 0 \qquad h_j(w) \neq 0$$

$$\theta_P(x) = \max_{\alpha, \beta : \alpha_i \geqslant 0} \left[f(x) + \sum_{i=1}^{k} \alpha_i c_i(x) + \sum_{j=1}^{l} \beta_j h_j(x) \right] = +\infty$$

拉格朗日对偶

$$\theta_p(x)=\begin{cases} f(x), & x\text{满足原始问题约束} \\ +\infty, & \text{其他} \end{cases}$$

- 考虑极小问题：

$$\min_x \theta_p(x)=\min_x \max_{\alpha,\beta:\alpha_i\geq 0} L(x,\alpha,\beta)$$

- 与原始最优化问题等价

$$p^*=\min_x \theta_p(x)$$

拉格朗日对偶

- 1、原始问题
  - 则：$$\min_{x} \theta_P(x) = \min_{x} \max_{\alpha,\beta:\alpha_i \geq 0} L(x,\alpha,\beta)$$
  - 称为广义拉格朗日函数的极小极大问题
  - 定义原始问题的最优值

$$p^* = \min_{x} \theta_P(x)$$

拉格朗日对偶

- 2、对偶问题
- 定义：$\theta_D(\alpha,\beta)=\min_x L(x,\alpha,\beta)$
- 则最大值问题：$\max_{\alpha,\beta:\alpha_i\geqslant 0}\theta_D(\alpha,\beta)=\max_{\alpha,\beta:\alpha_i\geqslant 0}\min_x L(x,\alpha,\beta)$
- 称为广义拉格朗日函数的极大极小问题
- 表示为约束最优化问题：

$$\max_{\alpha,\beta}\theta_D(\alpha,\beta)=\max_{\alpha,\beta}\min_x L(x,\alpha,\beta)$$

$$\text{s.t.}\quad \alpha_i\geqslant 0,\quad i=1,2,\cdots,k$$

- 称为原始问题的对偶问题，
- 对偶问题的最优值 $d^*=\max_{\alpha,\beta:\alpha_i\geqslant 0}\theta_D(\alpha,\beta)$

原始问题和对偶问题的关系

- **定理：**
- 若原始问题和对偶问题都有最优值，则

$$d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \min_{x} L(x,\alpha,\beta) \leqslant \min_{x} \max_{\alpha,\beta:\alpha_i \geq 0} L(x,\alpha,\beta) = p^*$$

- 推论：
- 设x*，和α*，β*分别是原始问题和对偶问题的可行解，并且d*=p*，则x*，和α*，β*分别是原始问题和对偶问题的最优解

原始问题和对偶问题的关系

- **定理：**
- 若原始问题和对偶问题都有最优值，则

$$d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \min_{x} L(x,\alpha,\beta) \leqslant \min_{x} \max_{\alpha,\beta:\alpha_i \geq 0} L(x,\alpha,\beta) = p^*$$

- 推论：
- 设x*，和α*，β*分别是原始问题和对偶问题的可行解，并且d*=p*，则x*，和α*，β*分别是原始问题和对偶问题的最优解

KKT条件

- **定理：**对原始问题和对偶问题，假设函数f(x)和$c_i(x)$是凸函数，$h_j(x)$是仿射函数，并且不等式$c_i(x)$是严格可行的，则x*，和α*，β*分别是原始问题和对偶问题的解的充分必要条件是x*，和α*，β*满足karush-Kuhn-Tucker(KKT)条件。

$$\nabla_x L(x^*,\alpha^*,\beta^*)=0$$

$$\nabla_\alpha L(x^*,\alpha^*,\beta^*)=0$$

$$\nabla_\beta L(x^*,\alpha^*,\beta^*)=0$$

$$\alpha_i^* c_i(x^*)=0,\quad i=1,2,\cdots,k$$

$$c_i(x^*)\leqslant 0,\quad i=1,2,\cdots,k$$

$$\alpha_i^*\geqslant 0,\quad i=1,2,\cdots,k$$

$$h_j(x^*)=0\quad j=1,2,\cdots,l$$

学习的对偶算法

- 对于线性可分支持向量机的优化问题，原始问题：

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2$$

$$\text{s.t.} \quad y_i(w \cdot x_i + b) - 1 \geqslant 0, \quad i = 1, 2, \cdots, N$$

1

- 应用拉格朗日对偶性，通过求解对偶问题，得到原始问题的解。
- 优点：
  - 对偶问题往往容易解
  - 引入核函数，推广到非线性分类问题

学习的对偶算法

- 定义拉格朗日函数

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

- 原问题：极小极大，对偶问题：极大极小

$$\min_x \theta_p(x)=\min_x \max_{\alpha,\beta:\alpha_i\geq 0} L(x,\alpha,\beta)$$

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

学习的对偶算法

- 先求L(w,b,α)对w，b的极小，再求对α的极大
- 1、求：$\min_{w,b} L(w,b,\alpha)$ ，对w，b分别求偏导并令等于o
- 由

$$\nabla_w L(w,b,\alpha) = w - \sum_{i=1}^{N} \alpha_i y_i x_i = 0 \quad \Rightarrow \quad w = \sum_{i=1}^{N} \alpha_i y_i x_i$$

$$\nabla_b L(w,b,\alpha) = \sum_{i=1}^{N} \alpha_i y_i = 0 \quad \Rightarrow \quad \sum_{i=1}^{N} \alpha_i y_i = 0$$

- 得：

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

$$= -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) + \sum_{i=1}^{N}\alpha_i \quad \Leftrightarrow \quad \min_{w,b} L(w,b,\alpha)$$

学习的对偶算法

- 求 $\min_{w,b} L(w,b,\alpha)$ 对α的极大，即是对偶问题：

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

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

$$\alpha_i \geqslant 0, \quad i = 1,2,\cdots,N$$

$$\Rightarrow \quad \min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^{N}\alpha_i$$

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

$$\alpha_i \geqslant 0, \quad i = 1,2,\cdots,N$$

学习的对偶算法

$$w^* = \sum_{i=1}^{N} \alpha_i^* y_i x_i \quad (7.25)$$

$$b^* = y_j - \sum_{i=1}^{N} \alpha_i^* y_i (x_i \cdot x_j) \quad (7.26)$$

定理：设 $\alpha^* = (\alpha_1^*, \alpha_2^*, \cdots, \alpha_l^*)^T$ 是对偶最优问题 2 的解

则存在下标j，使得 $\alpha_j^* > 0$ ，并可按下式求得原始问题 1 的解。

证明：由

$$\nabla_w L(w^*, b^*, \alpha^*) = w^* - \sum_{i=1}^{N} \alpha_i^* y_i x_i = 0 \quad (7.27)$$

$$\nabla_b L(w^*, b^*, \alpha^*) = -\sum_{i=1}^{N} \alpha_i^* y_i = 0$$

$$\alpha_i^* (y_i (w^* \cdot x_i + b^*) - 1) = 0, \quad i = 1, 2, \cdots, N$$

$$y_i (w^* \cdot x_i + b^*) - 1 \geqslant 0, \quad i = 1, 2, \cdots, N$$

$$\alpha_i^* \geqslant 0, \quad i = 1, 2, \cdots, N$$

得：$w^* = \sum_i \alpha_i^* y_i x_i$

学习的对偶算法

定理：设 $\alpha^* = (\alpha_1^*, \alpha_2^*, \cdots, \alpha_l^*)^T$ 是对偶最优问题 2 的解

则存在下标j，使得 $\alpha_j^* > 0$，并可按下式求得原始问题 1 的解。

证明：由 $w^* = \sum_i \alpha_i^* y_i x_i$，其中至少有一个 $\alpha_j^* > 0$

反证法：

假设：$\alpha^* = 0$ 由 (7.27) 可知w*=o，

但这不是原始优化问题的解，产生矛盾

对此：j 有 $y_j(w^* \cdot x_j + b^*) - 1 = 0$ (7.28)

将式 (7.25) 代入式 (7.28) 并注意到 $y_j^2 = 1$，

$$b^* = y_j - \sum_{i=1}^{N} \alpha_i^* y_i (x_i \cdot x_j)$$

学习的对偶算法

- 由此定理可知，分离超平面可以写成：

$$\sum_{i=1}^{N} \alpha_i^* y_i (x \cdot x_i) + b^* = 0$$

- 分类决策函数可以写成：

$$f(x) = \text{sign}\left(\sum_{i=1}^{N} \alpha_i^* y_i (x \cdot x_i) + b^* \right)$$

- 这就是说，分类决策函数只依赖于输入x和训练样本输入的内积，上式称为线性可分支持向量机的对偶形式.

线性可分支持向量机学习算法

- 输入：线性可分训练数据集 $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$

$$x_i \in \mathcal{X}=\mathbf{R}^n \qquad y_i \in \mathcal{Y}=\{-1,+1\}, \quad i=1,2,\cdots,N$$

- 输出：最大间隔分离超平面和分类决策函数
- 1、构造并求解约束最优化问题

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

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

$$\alpha_i \geqslant 0, \quad i=1,2,\cdots,N$$

求得最优解：$\alpha^*=(\alpha_1^*,\alpha_2^*,\cdots,\alpha_N^*)^{\mathrm{T}}$

线性可分支持向量机学习算法

- 2、计算 $$w^* = \sum_{i=1}^{N} \alpha_i^* y_i x_i$$
- 并选择α*的一个正分量 $\alpha_j^* > 0$，计算 $$b^* = y_j - \sum_{i=1}^{N} \alpha_i^* y_i (x_i \cdot x_j)$$
- 3、求得分离超平面 $$w^* \cdot x + b^* = 0$$
- 分类决策函数 $$f(x) = \text{sign}(w^* \cdot x + b^*)$$

支持向量

- 考虑原始优化问题和对偶优化问题，
- 将数据集中对应于 $\alpha_j^* > 0$ 的样本 (x_i, y_i) 的实例 $x_i \in \mathbf{R}^n$
- 称为支持向量

- 支持向量一定在分割边界上，由KKT互补条件：

$$\alpha_i^*(y_i(w^* \cdot x_i + b^*) - 1) = 0, \quad i = 1, 2, \cdots, N$$

- 对应于 $\alpha_j^* > 0$ 的样本 x_i

$$y_i(w^* \cdot x_i + b^*) - 1 = 0$$

- 或

$$w^* \cdot x_i + b^* = \pm 1$$

例子：

- 正例点 $x_1=(3,3)^T$，$x_2=(4,3)^T$　负例点 $x_3=(1,1)^T$
- 解：对偶形式

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

$$= \frac{1}{2}(18\alpha_1^2 + 25\alpha_2^2 + 2\alpha_3^2 + 42\alpha_1\alpha_2 - 12\alpha_1\alpha_3 - 14\alpha_2\alpha_3) - \alpha_1 - \alpha_2 - \alpha_3$$

$$\text{s.t.} \quad \alpha_1 + \alpha_2 - \alpha_3 = 0$$

$$\alpha_i \geqslant 0, \quad i = 1,2,3$$

- 将 $\alpha_3 = \alpha_1 + \alpha_2$ 带入目标函数并记为

$$s(\alpha_1,\alpha_2) = 4\alpha_1^2 + \frac{13}{2}\alpha_2^2 + 10\alpha_1\alpha_2 - 2\alpha_1 - 2\alpha_2$$

例子：

- 对 α_1,α_2 求偏导数，并令其为o，易知 $s(\alpha_1,\alpha_2)$ 在 $\left(\frac{3}{2},-1\right)^T$
- 取极值，但该点不满足约束条件 $\alpha_2 \geqslant 0$ ，所以最小值应在边界上达到
- 当 $\alpha_1=0$ 时，最小值 $s\left(0,\frac{2}{13}\right)=-\frac{2}{13}$
- 当 $\alpha_2=0$ 时，最小值 $s\left(\frac{1}{4},0\right)=-\frac{1}{4}$
- 于是 $s(\alpha_1,\alpha_2)$ 在 $\alpha_1=\frac{1}{4},\alpha_2=0$ 获得极小，$\alpha_3=\alpha_1+\alpha_2=\frac{1}{4}$
- 这样 $\alpha_1^*=\alpha_3^*=\frac{1}{4}$ 对应的实例向量为支持向量

例子：

- 计算得：

$$w_1^* = w_2^* = \frac{1}{2}$$

$$b^* = -2$$

- 分离超平面为：

$$\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2 = 0$$

- 分类决策函数为：

$$f(x) = \text{sign}\left(\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2\right)$$

二、线性支持向量机与软间隔最大化

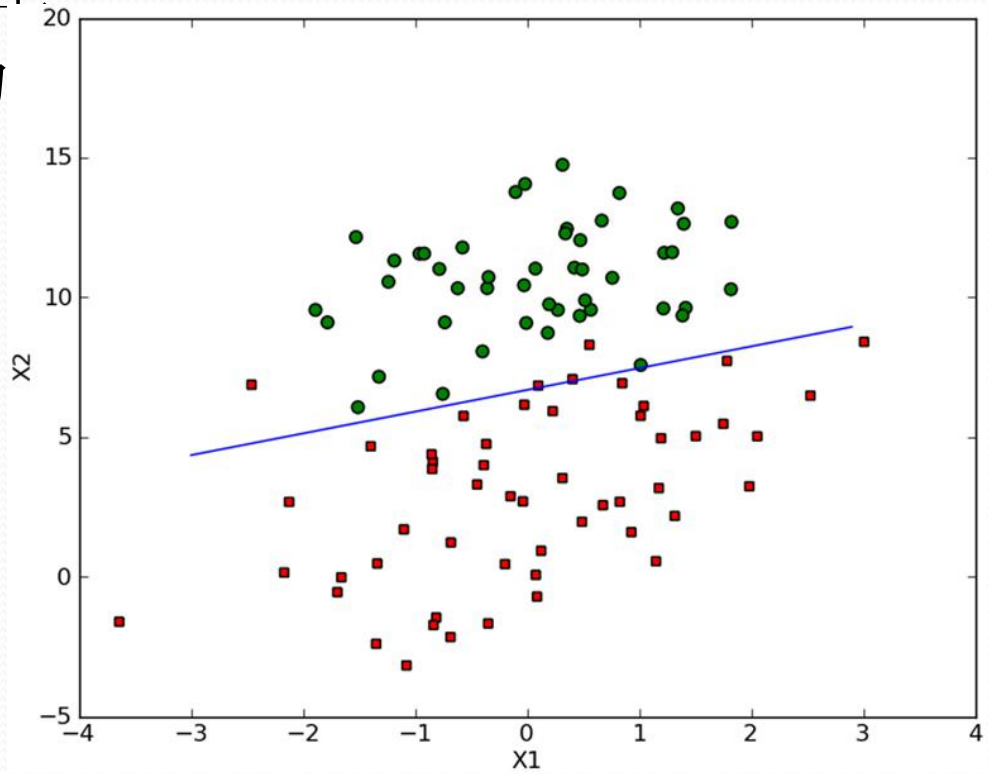
- 训练数据中有一些特异点（outlier），不能满足函数间隔大于等于1的约束条件。
- 解决方法：对每个样本点 (x_i, y_i) 引进一个松弛变量 $\xi_i \geq 0$
- 使得函数间隔加上松弛变量大于等于1，约束条件变为

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

目标函数变为：

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

C > 0 为惩罚参数



线性支持向量机与软间隔最大化

- 线性不可分的线性支持向量机的学习问题：

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

$$\text{s.t.} \quad y_i(w \cdot x_i + b) \geqslant 1 - \xi_i, \quad i = 1,2,\cdots,N \qquad (3)$$

$$\xi_i \geqslant 0, \quad i = 1,2,\cdots,N$$

- 可证明w的解是唯一的，b不是，
- 设该问题的解是w*,b*,可得到分离超平面和决策函数

$$w^* \cdot x + b^* = 0$$

$$f(x) = \text{sign}(w^* \cdot x + b^*)$$

线性支持向量机与软间隔最大化

- 原始问题 3 的拉格朗日函数：

$$L(w,b,\xi,\alpha,\mu)\equiv\frac{1}{2}\|w\|^2+C\sum_{i=1}^{N}\xi_i-\sum_{i=1}^{N}\alpha_i(y_i(w\cdot x_i+b)-1+\xi_i)-\sum_{i=1}^{N}\mu_i\xi_i$$

- 其中：$\alpha_i\geqslant 0,\mu_i\geqslant 0$
- 对偶问题是拉格朗日函数的极大极小问题
- 首先求 $L(w,b,\xi,\alpha,\mu)$ 对w,b,ξ的极小，由

$$\nabla_w L(w,b,\xi,\alpha,\mu)=w-\sum_{i=1}^{N}\alpha_i y_i x_i=0$$

$$\nabla_b L(w,b,\xi,\alpha,\mu)=-\sum_{i=1}^{N}\alpha_i y_i=0$$

$$\nabla_{\xi_i} L(w,b,\xi,\alpha,\mu)=C-\alpha_i-\mu_i=0$$

得：

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

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

$$C-\alpha_i-\mu_i=0$$

带

线性支持向量机与软间隔最大化

- 得：

$$\min_{w,b,\xi} L(w,b,\xi,\alpha,\mu) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) + \sum_{i=1}^{N}\alpha_i$$

- 再对 $\min_{w,b,\xi} L(w,b,\xi,\alpha,\mu)$ 求α的极大，得到对偶问题：

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

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

$$C - \alpha_i - \mu_i = 0 \quad \Rightarrow \quad 0 \leqslant \alpha_i \leqslant C$$

$$\alpha_i \geqslant 0$$

$$\mu_i \geqslant 0, \quad i = 1,2,\cdots,N$$

线性支持向量机与软间隔最大化

- 原始问题 3 的对偶问题：

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

$$\text{s.t.} \quad \sum_{i=1}^{N}\alpha_i y_i = 0 \qquad 4$$

$$0 \leqslant \alpha_i \leqslant C, \quad i = 1,2,\cdots,N$$

- 定理：
- 设 $\alpha^* = (\alpha_1^*, \alpha_2^*, \cdots, \alpha_N^*)^T$ 是对偶问题 4 的一个解，若存在α*的一个分量 α_j^*，$0 < \alpha_j^* < C$，则原始问题的解w*,b*

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

$$b^* = y_j - \sum_{i=1}^{N} y_i \alpha_i^* (x_i \cdot x_j)$$

线性支持向量机学习算法

- 输入：线性不可分训练数据集 $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$

$$x_i \in \mathcal{X}=\mathbf{R}^n \qquad y_i \in \mathcal{Y}=\{-1,+1\}, \quad i=1,2,\cdots,N$$

- 输出：分离超平面和分类决策函数
- 1、构造并求解约束最优化问题

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

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

$$0 \leqslant \alpha_i \leqslant C, \quad i=1,2,\cdots,N$$

求得最优解：

$$\alpha^* = (\alpha_1^*, \alpha_2^*, \cdots, \alpha_N^*)^{\mathrm{T}}$$

线性支持向量机学习算法

- 2、计算 $$w^* = \sum_{i=1}^{N} \alpha_i^* y_i x_i$$
- 并选择α*，适合条件 $0 < \alpha_j^* < C$ ，计算 $$b^* = y_j - \sum_{i=1}^{N} \alpha_i^* y_i (x_i \cdot x_j)$$
- 3、求得分离超平面 $$w^* \cdot x + b^* = 0$$
- 分类决策函数 $$f(x) = \text{sign}(w^* \cdot x + b^*)$$

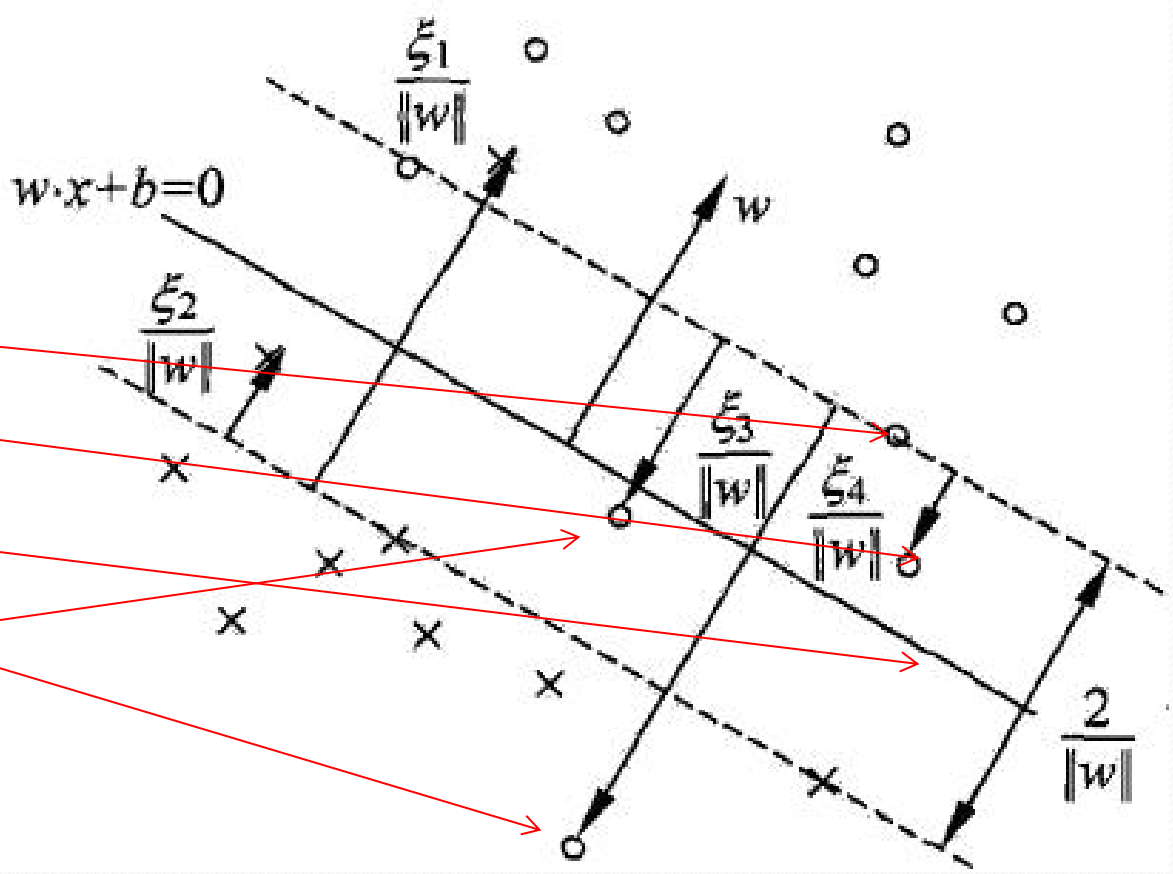
支持向量

若 $\alpha_i^* < C$，则 $\xi_i = 0$

若 $\alpha_i^* = C$，$0 < \xi_i < 1$

若 $\alpha_i^* = C$，$\xi_i = 1$

若 $\alpha_i^* = C$，$\xi_i > 1$



合页损失函数hinge loss function

- 线性支持向量机学习还有另外一种解释，就是最小化以下目标函数：

$$\sum_{i=1}^{N}[1-y_i(w\cdot x_i+b)]_+ +\lambda\|w\|^2$$

- 第一项：$L(y(w\cdot x+b))=[1-y(w\cdot x+b)]_+$
- 称为合页损失函数

$$[z]_+=\begin{cases}z, & z>0\\ 0, & z\leqslant 0\end{cases}$$

合页损失函数hinge loss function

- 线性支持向量机原始最优化问题:

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

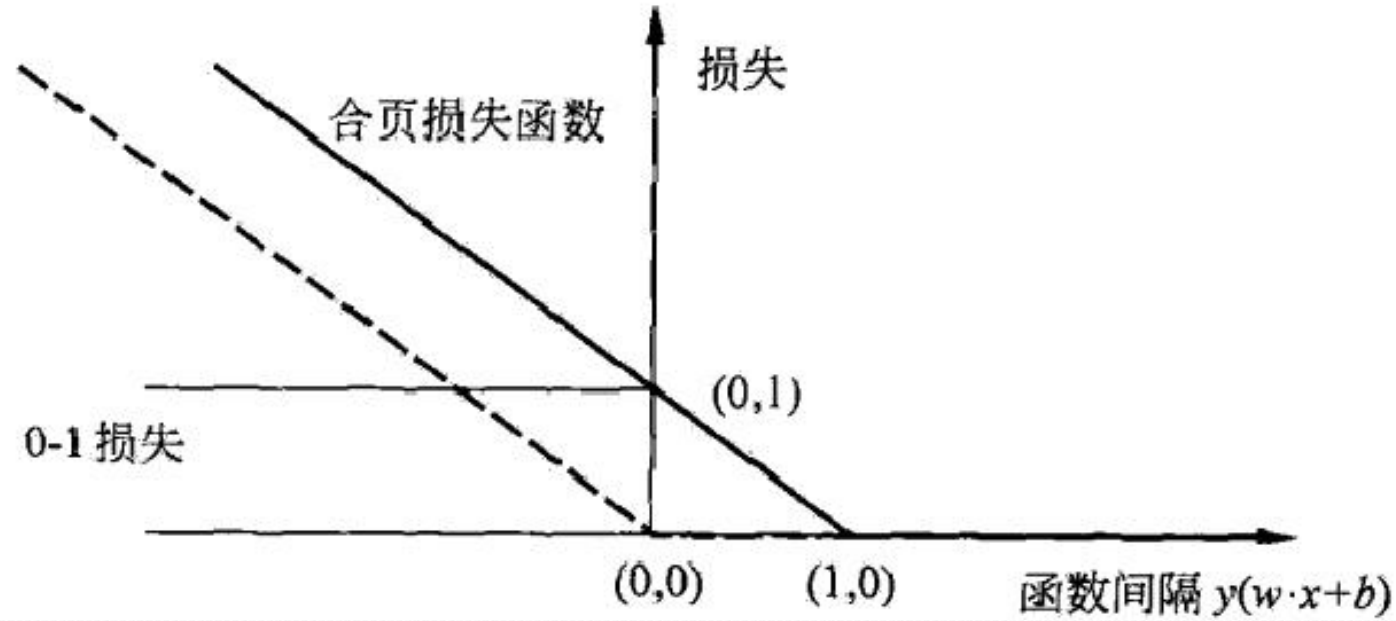
$$\text{s.t.} \quad y_i(w \cdot x_i + b) \geqslant 1 - \xi_i, \quad i = 1, 2, \cdots, N$$

$$\xi_i \geqslant 0, \quad i = 1, 2, \cdots, N$$

- 等价于:

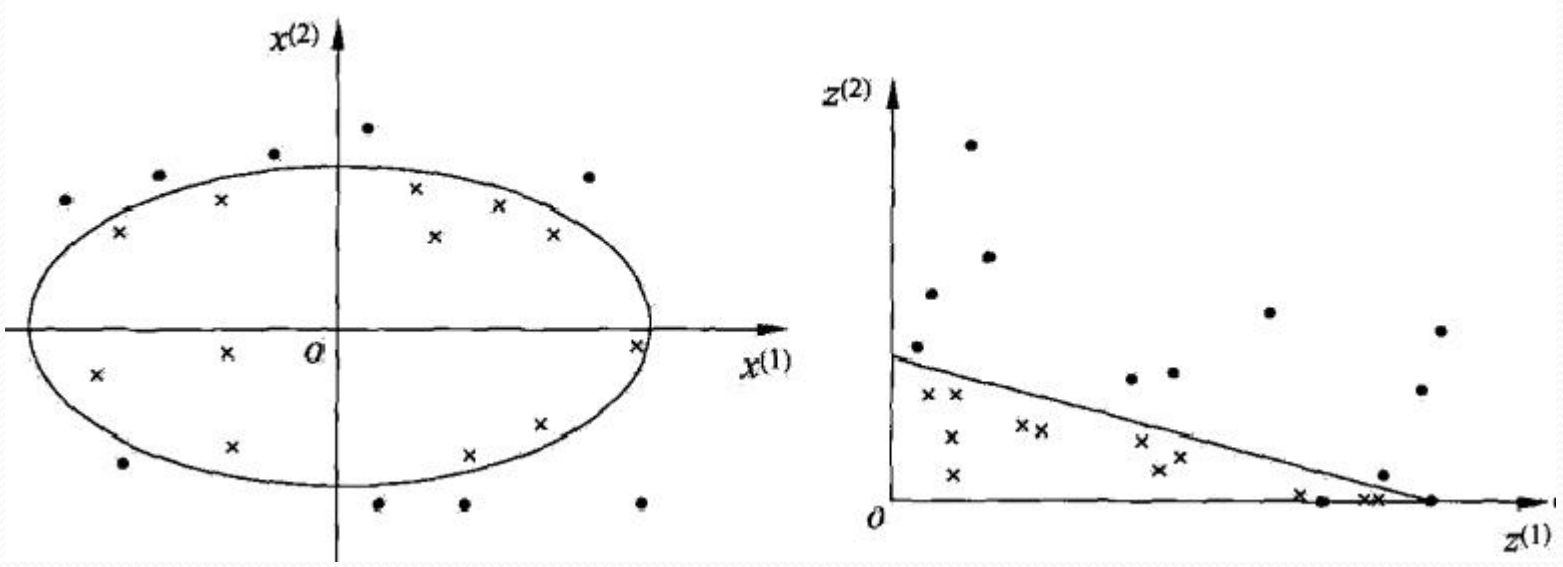
$$\min_{w,b} \quad \sum_{i=1}^{N}\left[1 - y_i(w \cdot x_i + b)\right]_+ + \lambda\|w\|^2$$

合页损失函数hinge loss function



三、非线性支持向量机与核函数

- 非线性分类问题：
- 非线性可分问题
  - 如果能用R^n中的一个超曲面将正负例正确分开，则称这个问题为非线性可分问题.



非线性支持向量机与核函数

- 非线性问题往往不好求解，所以希望能用解线性分类间题的方法解决这个问题。
- 采取的方法是进行一个非线性变换，将非线性问题变换为线性问题，通过解变换后的线性问题的方法求解原来的非线性问题。
- 原空间：

$$\mathcal{X} \subset \mathbf{R}^2, x = (x^{(1)}, x^{(2)})^{\mathrm{T}} \in \mathcal{X}$$

- 新空间：

$$\mathcal{Z} \subset \mathbf{R}^2, z = (z^{(1)}, z^{(2)})^{\mathrm{T}} \in \mathcal{Z} \qquad z = \phi(x) = ((x^{(1)})^2, (x^{(2)})^2)^{\mathrm{T}}$$

$$w_1(x^{(1)})^2 + w_2(x^{(2)})^2 + b = 0 \Rightarrow w_1 z^{(1)} + w_2 z^{(2)} + b = 0$$

非线性支持向量机与核函数

- 用线性分类方法求解非线性分类问题分为两步:
  - 首先使用一个变换将原空间的数据映射到新空间;
  - 然后在新空间里用线性分类学习方法从训练数据中学习分类模型。
- 核技巧就属于这样的方法
  - 核技巧应用到支持向量机，其基本想法：
  - 通过一个非线性变换将输入空间(欧氏空间R”或离散集合)对应于一个特征空间(希尔伯特空间)，使得在输入空间中的超曲面模型对应于特征空间中的超平面模型(支持向量机)。分类问题的学习任务通过在特征空间中求解线性支持向量机就可以完成.

非线性支持向量机与核函数

- 核函数定义：
- 设X是输入空间(欧氏空间R^n的子集或离散集合)，又设H为特征空间(希尔伯特空间)，如果存在一个从X到H的映射

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

- 使得对所有 $x, z \in \mathcal{X}$

- 函数K(x,z)满足条件 $K(x,z) = \phi(x) \cdot \phi(z)$

- 则称 $K(x,z)$ 为核函数，$\phi(x)$ 为映射函数，
- 式中 $\phi(x) \cdot \phi(z)$ 为 $\phi(x)$ 和 $\phi(z)$ 的内积

非线性支持向量机与核函数

- 核技巧的想法是：
- 在学习与预测中只定义核函数K(x,z)，而不显式地定义映射函数，通常，直接计算K(x,z)比较容易，而通过$\phi(x)$和$\phi(z)$计算K(x, z)并不容易。
- 注意：φ是 输入空间R^n到特征空间H的映射，特征空间H一般是高维，映射可以不同。

非线性支持向量机与核函数

- 例：
- 假设输入空间是R^2，核函数是$K(x,z)=(x\cdot z)^2$,试找出其相关的特征空间H和映射 $\phi(x):\mathbf{R}^2\to\mathcal{H}$
- 解：

取特征空间 $\mathcal{H}=\mathbf{R}^3$， 记 $x=(x^{(1)},x^{(2)})^{\mathrm{T}}$， $z=(z^{(1)},z^{(2)})^{\mathrm{T}}$

$$(x\cdot z)^2=(x^{(1)}z^{(1)}+x^{(2)}z^{(2)})^2=(x^{(1)}z^{(1)})^2+2x^{(1)}z^{(1)}x^{(2)}z^{(2)}+(x^{(2)}z^{(2)})^2$$

- 可以取： $\phi(x)=((x^{(1)})^2,\sqrt{2}x^{(1)}x^{(2)},(x^{(2)})^2)^{\mathrm{T}}$

- 容易验证： $\phi(x)\cdot\phi(z)=(x\cdot z)^2=K(x,z)$

非线性支持向量机与核函数

- 例：
- 假设输入空间是R^2，核函数是$K(x,z)=(x\cdot z)^2$,试找出其相关的特征空间H和映射 $\phi(x):\mathbf{R}^2\to\mathcal{H}$
- 解：
- 同样：

$$\phi(x)=\frac{1}{\sqrt{2}}((x^{(1)})^2-(x^{(2)})^2,2x^{(1)}x^{(2)},(x^{(1)})^2+(x^{(2)})^2)^T$$

$$\phi(x)=((x^{(1)})^2,x^{(1)}x^{(2)},x^{(1)}x^{(2)},(x^{(2)})^2)^T$$

- 都满足条件。

核函数在支持向量机的应用

- 注意到：
- 线性支持向量机对偶问题中，无论是目标函数还是决策函数都只涉及输入实例和实例之间的内积。
- 目标函数中的内积 $x_i \cdot x_j$ 用核函数 $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ 代替，目标函数：

$$W(\alpha) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^{N}\alpha_i$$

- 决策函数：

$$f(x) = \text{sign}\left(\sum_{i=1}^{N_s} a_i^* y_i \phi(x_i) \cdot \phi(x) + b^*\right) = \text{sign}\left(\sum_{i=1}^{N_s} a_i^* y_i K(x_i, x) + b^*\right)$$

正定核

- 问题：
  - 已知映射函数φ，可以通过 $\phi(x)$ 和 $\phi(z)$ 的内积求得核函数K(x,z).
  - 不用构造映射φ，能否直接判断一个给定的函数K(x,z)是不是核函数?
  - 或者说，函数K(x,z)满足什么条件才能成为核函数?
- 假设K(x,z)是定义在XxX上的对称函数，并且对任意的

$$x_1, x_2, \cdots, x_m \in \mathcal{X}$$

- K(x,z)关于 $x_1, x_2, \cdots, x_m$ 的Gram矩阵是半正定的，可以依据函数K(x,z)，构成一个希尔伯特空间(Hilbert space)；
- 其步骤是首先定义映射φ，并构成向量空间S，然后在S上定义内积构成内积空间; 最后将S完备化构成希尔伯特空间.

正定核

- 1、定义映射，构成向量空间S
- 映射：$\phi : x \to K(\cdot, x)$
- 对任意 $x_i \in \mathcal{X}$， $\alpha_i \in \mathbf{R}$， $i = 1, 2, \cdots, m$
- 定义线性组合：$$f(\cdot) = \sum_{i=1}^{m} \alpha_i K(\cdot, x_i)$$
- 考虑由线性组合为元素的集合S, 由于集合S对加法和数乘运算是封闭的，S构成一个向量空间。

正定核

- 2、在S上定义内积，构成内积空间
- 在S上定义一个运算“*”，对任意f，g属于S

$$f(\cdot)=\sum_{i=1}^{m}\alpha_i K(\cdot,x_i)$$

$$g(\cdot)=\sum_{i=1}^{l}\beta_j K(\cdot,z_j)$$

- 定义运算*：

$$f*g=\sum_{i=1}^{m}\sum_{j=1}^{l}\alpha_i\beta_j K(x_i,z_j)$$

- 证明内积空间：

（1） $(cf)*g=c(f*g)$， $c\in\mathbf{R}$

（2） $(f+g)*h=f*h+g*h$， $h\in S$

（3） $f*g=g*f$

（4） $f*f\geqslant 0$， $f*f=0\Leftrightarrow f=0$

正定核

- 3、将内积空间S完备化为希尔伯特空间
- 由：$$f \cdot g = \sum_{i=1}^{m} \sum_{j=1}^{l} \alpha_i \beta_j K(x_i, z_j)$$
- 内积得到范数：$$\| f \| = \sqrt{f \cdot f}$$
- 因此，S是一个赋范向量空间；根据泛函分析理论，对于不完备的赋范向量空间S，一定可以使之完备化，得到完备的赋范向量空间H；一个内积空间，当作为一个赋范向量空间是完备的时候，就是希尔伯特空间，这样，就得到了希尔伯特空间H。
- 再生性：

$$K(\cdot, x) \cdot f = f(x) \qquad K(\cdot, x) \cdot K(\cdot, z) = K(x, z)$$

正定核

- 正定核的充要条件
- 设K: $\mathcal{X}\times\mathcal{X}\to\mathbf{R}$ ，是对称函数，则K(x,z)为正定核函数的充要条件是对任意 $x_i\in\mathcal{X}$， $i=1,2,\cdots,m$， K(x,z)对应的Gram矩阵

$$K=\left[K(x_i,x_j)\right]_{m\times m}$$

- 是半正定的。
- 给定一个实矩阵 A，矩阵 $A^{\mathrm{T}}A$ 是 A 的列向量的格拉姆矩阵，而矩阵 AA^{T} 是 A 的行向量的格拉姆矩阵。
- 格拉姆矩阵是半正定的，反之每个半正定矩阵是某些向量的格拉姆矩阵。这组向量一般不是惟一的：任何正交基的格拉姆矩阵是恒同矩阵。

正定核

- 正定核的等价定义
- 设$\mathcal{X} \subset \mathbf{R}^n$，是K(x,z)是定义在 $\mathcal{X} \times \mathcal{X}$ 对称函数，如果对任意的 $x_i \in \mathcal{X}$， $i=1,2,\cdots,m$， K(x,z)对应的Gram矩阵

$$K=\left[K(x_i,x_j)\right]_{m\times m}$$

- 半正定的，则称K(x,z)为正定核。

- 这一定义在构造核函数时很有用。但对于一个具体函数K(x,z) 来说，检验它是否为$\{x_1,x_2,\cdots,x_m\}$并不容易，因为要求对任意有限输入集 验证K对应的Gram矩阵是否为半正定的。
- 在实际问题中往往应用已有的核函数。

常用核函数

- 1、多项式核函数（Polynomial kernel function)

$$K(x,z)=(x\cdot z+1)^p$$

- 对应的支持向量机为P次多项式分类器，分类决策函数：

$$f(x)=\text{sign}\left(\sum_{i=1}^{N_s}a_i^*y_i(x_i\cdot x+1)^p+b^*\right)$$

- 2、高斯核函数 （Gaussian Kernel Function)

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

- 决策函数：

$$f(x)=\text{sign}\left(\sum_{i=1}^{N_s}a_i^*y_i\exp\left(-\frac{\|x-z\|^2}{2\sigma^2}\right)+b^*\right)$$

常用核函数

- 3、字符串核函数：

非线性支持向量机学习算法

- 输入：线性不可分训练数据集 $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$

$$x_i \in \mathcal{X}=\mathbf{R}^n \qquad y_i \in \mathcal{Y}=\{-1,+1\}, \quad i=1,2,\cdots,N$$

- 输出：分类决策函数
- 1、选取适当的核函数和参数C，构造最优化问题：

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(x_i,x_j)-\sum_{i=1}^{N}\alpha_i$$

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

$$0\leqslant\alpha_i\leqslant C, \quad i=1,2,\cdots,N$$

求得最优解：$\alpha^*=(\alpha_1^*,\alpha_2^*,\cdots,\alpha_N^*)^{\mathrm{T}}$

非线性支持向量机学习算法

- 2、并选择α*，适合条件 $0<\alpha_j^*<C$ ，计算

$$b^* = y_j - \sum_{i=1}^{N} \alpha_i^* y_i K(x_i \cdot x_j)$$

- 3、构造决策函数

$$f(x) = \text{sign}\left(\sum_{i=1}^{N} \alpha_i^* y_i K(x \cdot x_i) + b^* \right)$$

- 当K(x,z)是正定核函数时，5 是凸二次规划问题，解是存在的。

四、序列最小最优化算法

- 序列最小最优化(sequential minimal optimization SMO)算法：1998年由Platt提出。

John C. Platt, "Using Analytic QP and Sparseness to Speed Training of Support Vector Machines" in *Advances in Neural Information Processing Systems* 11, M. S. Kearns, S. A. Solla, D. A. Cohn, eds (MIT Press, 1999), 557–63.

- 动机：
- 支持向量机的学习问题可以形式化为求解凸二次规划问题.这样的凸二次规划问题具有全局最优解，并且有许多最优化算法可以用于这一问题的求解；
- 但是当训练样本容量很大时，这些算法往往变得非常低效，以致无法使用.所以，如何高效地实现支持向量机学习就成为一个重要的问题。

序列最小最优化算法

- SMO（Sequential minimal optimization）
- 解如下凸二次规划的对偶问题

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j K(x_i,x_j)-\sum_{i=1}^{N}\alpha_i$$

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

$$0 \leqslant \alpha_i \leqslant C, \quad i=1,2,\cdots,N$$

- 注意：变量是拉格朗日乘子α_i，一个对应一个样本

SMO算法

- 启发式算法，基本思路：
- 如果所有变量的解都满足此最优化问题的KKT条件，那么得到解；
- 否则，选择两个变量，固定其它变量，针对这两个变量构建一个二次规划问题，称为子问题，可通过解析方法求解，提高了计算速度。
- 子问题的两个变量：一个是违反KKT条件最严重的那个，另一个由约束条件自动确定。

$$\alpha_1 = -y_1 \sum_{i=2}^{N} \alpha_i y_i$$

- SMO算法包括两个部分：
  - 求解两个变量二次规划的解析方法
  - 选择变量的启发式方法

两个变量二次规划的求解过程

- 选择两个变量，其它固定，SMO的 5 的子问题：

$$\min_{\alpha_1,\alpha_2} \quad W(\alpha_1,\alpha_2)=\frac{1}{2}K_{11}\alpha_1^2+\frac{1}{2}K_{22}\alpha_2^2+y_1y_2K_{12}\alpha_1\alpha_2$$

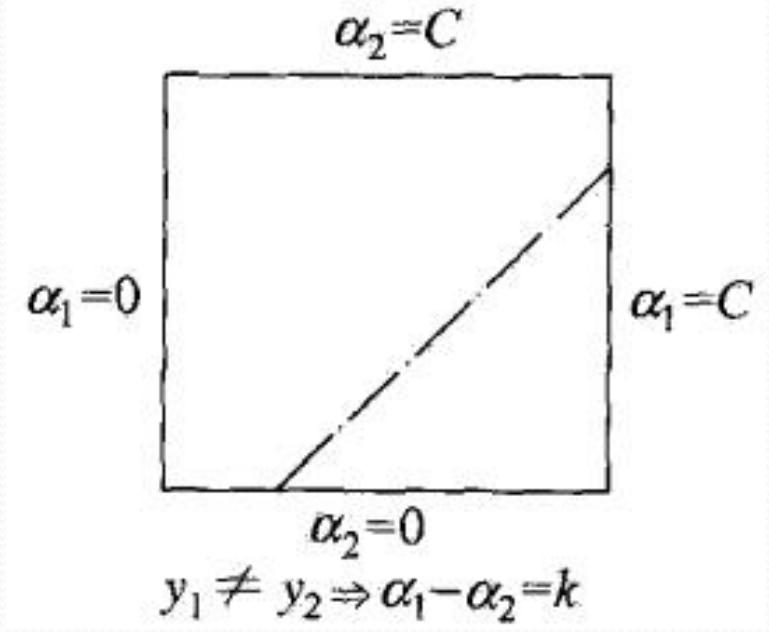
$$-(\alpha_1+\alpha_2)+y_1\alpha_1\sum_{i=3}^{N}y_i\alpha_iK_{i1}+y_2\alpha_2\sum_{i=3}^{N}y_i\alpha_iK_{i2} \qquad 6$$

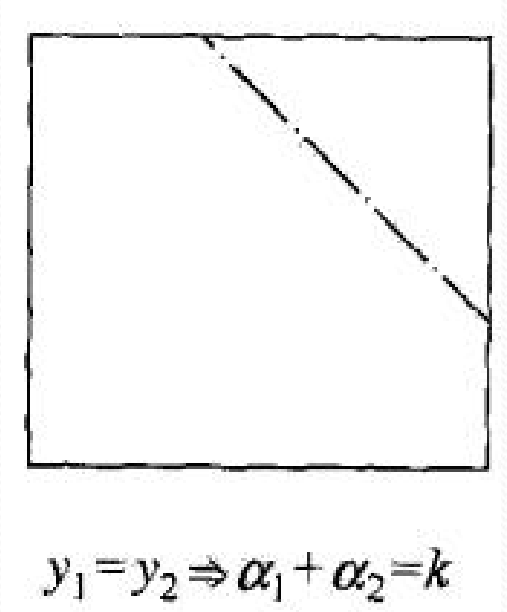
$$\text{s.t.} \quad \alpha_1y_1+\alpha_2y_2=-\sum_{i=3}^{N}y_i\alpha_i=\varsigma$$

$$0\leqslant\alpha_i\leqslant C, \quad i=1,2$$

两个变量二次规划的求解过程

- 两个变量，约束条件用二维空间中的图形表示





- 假设问题 6 的初始可行解为 $\alpha_1^{old}, \alpha_2^{old}$，最优解 $\alpha_1^{new}, \alpha_2^{new}$
- 设α_2未经剪辑时的最优解为 $\alpha_2^{new,unc}$

两个变量二次规划的求解过程

- 根据不等式条件 α_2^{new} 的取值范围：

$$L \leqslant \alpha_2^{new} \leqslant H$$

- 左图：$L = \max(0, \alpha_2^{old} - \alpha_1^{old})$ $H = \min(C, C + \alpha_2^{old} - \alpha_1^{old})$

- 右图：$L = \max(0, \alpha_2^{old} + \alpha_1^{old} - C)$ $H = \min(C, \alpha_2^{old} + \alpha_1^{old})$

两个变量二次规划的求解过程

- 求解过程：
- 先求沿着约束方向未经剪辑时的 $\alpha_2^{new,unc}$
- 再求剪辑后的 α_2^{new}

- 记： $g(x)=\sum_{i=1}^{N}\alpha_i y_i K(x_i,x)+b$

- 令： $E_i=g(x_i)-y_i=\left(\sum_{j=1}^{N}\alpha_j y_j K(x_j,x_i)+b\right)-y_i,\quad i=1,2$

- E 为输入x的预测值和真实输出y的差，i=1，2

两个变量二次规划的求解过程

- 定理：
- 最优化问题 6 沿约束方向未经剪辑的解：

$$\alpha_2^{new,unc} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{\eta}$$

$$\eta = K_{11} + K_{22} - 2K_{12} = \left\| \Phi(x_1) - \Phi(x_2) \right\|^2$$

- 剪辑后的解

$$\alpha_2^{new} = \begin{cases} H, & \alpha_2^{new,unc} > H \\ \alpha_2^{new,unc}, & L \leqslant \alpha_2^{new,unc} \leqslant H \\ L, & \alpha_2^{new,unc} < L \end{cases}$$

- 得到α_1的解

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

两个变量二次规划的求解过程

- 证明： 引进记号

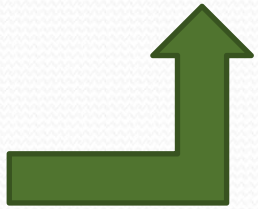
$$v_i = \sum_{j=3}^{N} \alpha_j y_j K(x_i, x_j) = g(x_i) - \sum_{j=1}^{2} \alpha_j y_j K(x_i, x_j) - b, \quad i = 1,2$$

- 目标函数写成：

$$W(\alpha_1, \alpha_2) = \frac{1}{2} K_{11} \alpha_1^2 + \frac{1}{2} K_{22} \alpha_2^2 + y_1 y_2 K_{12} \alpha_1 \alpha_2 - (\alpha_1 + \alpha_2) + y_1 v_1 \alpha_1 + y_2 v_2 \alpha_2$$

- 由 $\alpha_1 y_1 = \varsigma - \alpha_2 y_2$ 及 $y_i^2 = 1$

$$\alpha_1 = (\varsigma - y_2 \alpha_2) y_1$$



两个变量二次规划的求解过程

- 得到只是α2 的函数的目标函数

$$W(\alpha_2)=\frac{1}{2}K_{11}(\varsigma-\alpha_2 y_2)^2+\frac{1}{2}K_{22}\alpha_2^2+y_2K_{12}(\varsigma-\alpha_2 y_2)\alpha_2$$
$$-(\varsigma-\alpha_2 y_2)y_1-\alpha_2+v_1(\varsigma-\alpha_2 y_2)+y_2v_2\alpha_2$$

- 对α2求导

$$\frac{\partial W}{\partial \alpha_2}=K_{11}\alpha_2+K_{22}\alpha_2-2K_{12}\alpha_2-K_{11}\varsigma y_2+K_{12}\varsigma y_2+y_1y_2-1-v_1y_2+y_2v_2$$

- 令其为0：

$$(K_{11}+K_{22}-2K_{12})\alpha_2=y_2(y_2-y_1+\varsigma K_{11}-\varsigma K_{12}+v_1-v_2)$$
$$=y_2\left[y_2-y_1+\varsigma K_{11}-\varsigma K_{12}+\left(g(x_1)-\sum_{j=1}^{2}y_j\alpha_jK_{1j}-b\right)-\left(g(x_2)-\sum_{j=1}^{2}y_j\alpha_jK_{2j}-b\right)\right]$$

两个变量二次规划的求解过程

- 将 $\varsigma = \alpha_1^{old} y_1 + \alpha_2^{old} y_2$ 代入：

$$
\begin{aligned}
(K_{11} + K_{22} - 2K_{12})\alpha_2^{new,unc} &= y_2((K_{11} + K_{22} - 2K_{12})\alpha_2^{old} y_2 + y_2 - y_1 + g(x_1) - g(x_2)) \\
&= (K_{11} + K_{22} - 2K_{12})\alpha_2^{old} + y_2(E_1 - E_2)
\end{aligned}
$$

- 将 $\eta = K_{11} + K_{22} - 2K_{12}$ 代入：

$$
\alpha_2^{new,unc} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{\eta}
$$

两个变量二次规划的求解过程

- 得到定理：
- 最优化问题 6 沿约束方向未经剪辑的解：

$$\alpha_2^{new,unc} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{\eta}$$

$$\eta = K_{11} + K_{22} - 2K_{12} = \left\| \Phi(x_1) - \Phi(x_2) \right\|^2$$

- 剪辑后的解

$$\alpha_2^{new} = \begin{cases} H, & \alpha_2^{new,unc} > H \\ \alpha_2^{new,unc}, & L \leqslant \alpha_2^{new,unc} \leqslant H \\ L, & \alpha_2^{new,unc} < L \end{cases}$$

- 得到α_1的解

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

变量的选择方法

- SMO算法在每个子问题中选择两个变量优化，其中至少一个变量是违反KKT条件的
- 1、第一个变量的选择：外循环
- 违反KKT最严重的样本点，
- 检验样本点是否满足KKT条件：

$$\alpha_i = 0 \Leftrightarrow y_i g(x_i) \geqslant 1$$

先检查 → $0 < \alpha_i < C \Leftrightarrow y_i g(x_i) = 1$

$$\alpha_i = C \Leftrightarrow y_i g(x_i) \leqslant 1$$

$$g(x_i) = \sum_{j=1}^{N} \alpha_j y_j K(x_i, x_j) + b$$

变量的选择方法

- 2、第二个变量的检查：内循环，
  - 选择的标准是希望能使目标函数有足够大的变化
  - 即对应$|E_1 - E_2|$最大，即E1，E2 的符号相反，差异最大
  - 如果内循环通过上述方法找到的点不能使目标函数有足够的下降
  - 则：遍历间隔边界上的样本点，测试目标函数下降
  - 如果下降不大，则遍历所有样本点
  - 如果依然下降不大，则丢弃外循环点，重新选择

计算阈值b和Ei

- 3、每次完成两个变量的优化后，重新计算b，Ei
- 由KKT条件：

$$\sum_{i=1}^{N}\alpha_i y_i K_{i1} + b = y_1$$

$$b_1^{new} = y_1 - \sum_{i=3}^{N}\alpha_i y_i K_{i1} - \alpha_1^{new} y_1 K_{11} - \alpha_2^{new} y_2 K_{21}$$

$$E_i = g(x_i) - y_i = \left(\sum_{j=1}^{N}\alpha_j y_j K(x_j, x_i) + b\right) - y_i, \quad i = 1,2$$

$$E_1 = \sum_{i=3}^{N}\alpha_i y_i K_{i1} + \alpha_1^{old} y_1 K_{11} + \alpha_2^{old} y_2 K_{21} + b^{old} - y_1$$

计算阈值b和Ei

- 3、每次完成两个变量的优化后，重新计算b，Ei
- 由KKT条件：

$$\sum_{i=1}^{N}\alpha_i y_i K_{i1} + b = y_1$$

$$b_1^{new} = y_1 - \sum_{i=3}^{N}\alpha_i y_i K_{i1} - \alpha_1^{new} y_1 K_{11} - \alpha_2^{new} y_2 K_{21}$$

$$E_1 = \sum_{i=3}^{N}\alpha_i y_i K_{i1} + \alpha_1^{old} y_1 K_{11} + \alpha_2^{old} y_2 K_{21} + b^{old} - y_1$$

$$y_1 - \sum_{i=3}^{N}\alpha_i y_i K_{i1} = -E_1 + \alpha_1^{old} y_1 K_{11} + \alpha_2^{old} y_2 K_{21} + b^{old}$$

$$b_1^{new} = -E_1 - y_1 K_{11}(\alpha_1^{new} - \alpha_1^{old}) - y_2 K_{21}(\alpha_2^{new} - \alpha_2^{old}) + b^{old}$$

计算阈值b和E_i

- 如果：$0<\alpha_2^{new}<C$

$$b_2^{new}=-E_2-y_1K_{12}(\alpha_1^{new}-\alpha_1^{old})-y_2K_{22}(\alpha_2^{new}-\alpha_2^{old})+b^{old}$$

$$E_i^{new}=\sum_S y_j\alpha_j K(x_i,x_j)+b^{new}-y_i$$

S 是所有支持向量 x_j 的集合

SMO算法

- 输入：训练数据集 $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$
- $x_i\in\mathcal{X}=\mathbf{R}^n$　$y_i\in\mathcal{Y}=\{-1,+1\}$，$i=1,2,\cdots,N$，精度ε
- 输出：近似解 α

（1）取初值 $\alpha^{(0)}=0$，令 $k=0$

（2）选取优化变量 $\alpha_1^{(k)},\alpha_2^{(k)}$，解析求解两个变量的最优化问题

求得最优解 $\alpha_1^{(k+1)},\alpha_2^{(k+1)}$，更新 α 为 $\alpha^{(k+1)}$；

（3）若在精度 ε 范围内满足停机条件

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

$$0\leqslant\alpha_i\leqslant C,\quad i=1,2,\cdots,N$$

$$y_i\cdot g(x_i)=\begin{cases}\geqslant 1, & \{x_i\mid\alpha_i=0\}\\ =1, & \{x_i\mid 0<\alpha_i<C\}\\ \leqslant 1, & \{x_i\mid\alpha_i=C\}\end{cases}$$

$$g(x_i)=\sum_{j=1}^{N}\alpha_j y_j K(x_j,x_i)+b$$

则转 (4)；否则令 $k=k+1$，转 (2)；

（4）取 $\hat{\alpha}=\alpha^{(k+1)}$

SMO算法

- SVM light：Joachims
- http://svmlight.joachims.org
- LIBSVM:http://www.csie.ntu.edu.tw/~cjlin/libsvm

Q & A