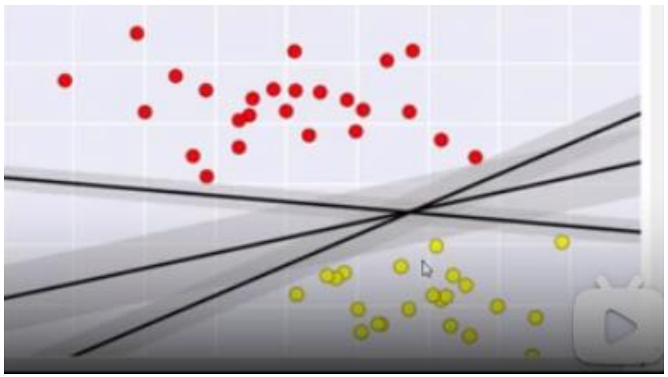
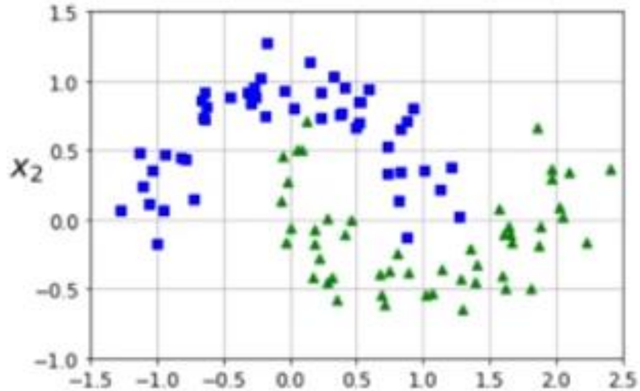
第6章 支持向量机(SVM)

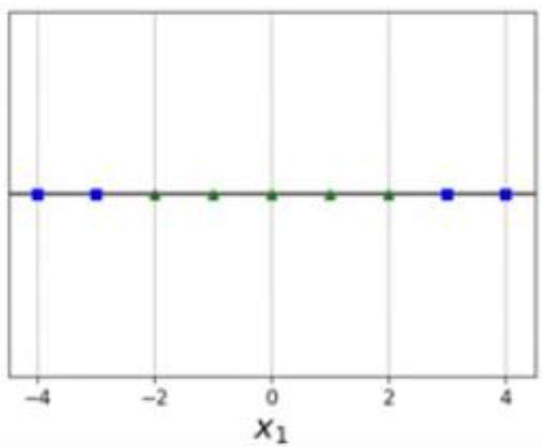
Support Vector Machine

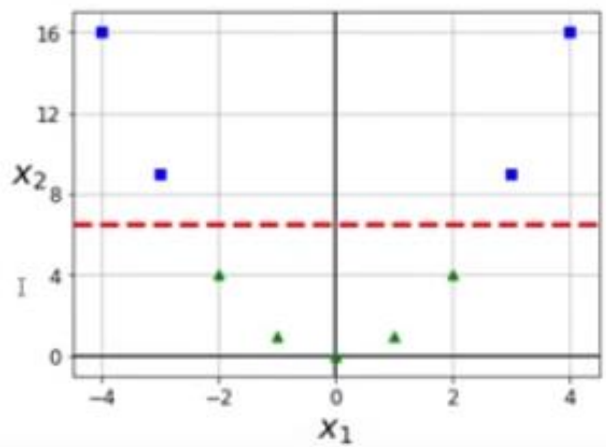
一、支持向量机解决什么问题?

- 要解决的问题：什么样的决策边界才是最好的呢？
- 特征数据本身如果就很难分，怎么办呢？
- 计算复杂度怎么样？能实际应用吗？







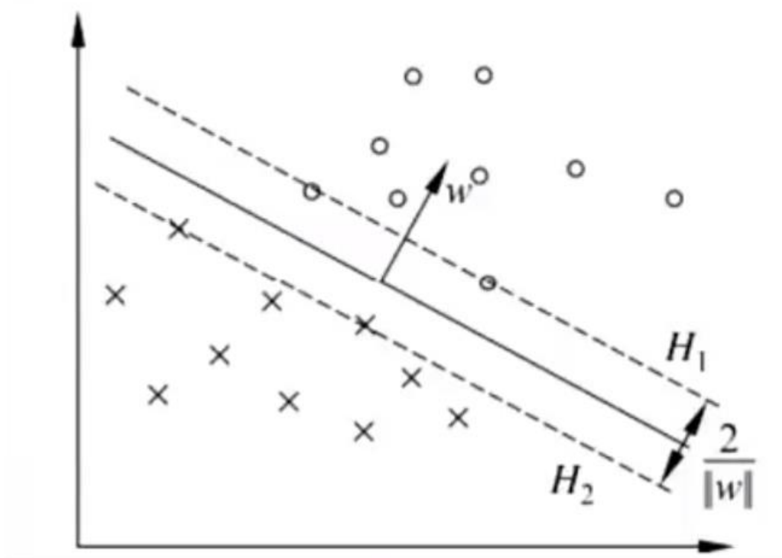


一、线性可分支持向量机

1、线性可分问题

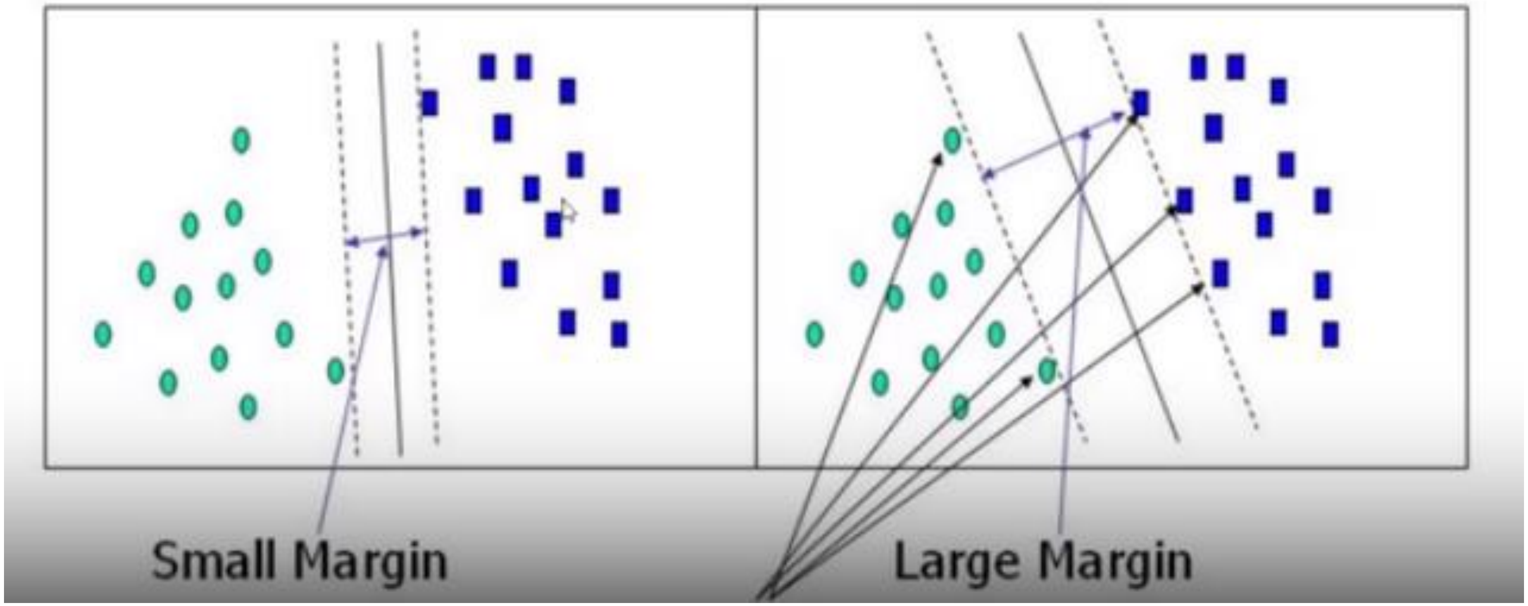
$$\mathbf{w}^T\mathbf{x} + b = 0$$

$$f(x) = \operatorname{sgn}\left(w^{*T}x + b^*\right)$$



一、线性可分支持向量机

2、最大间隔超平面



一、线性可分支持向量机

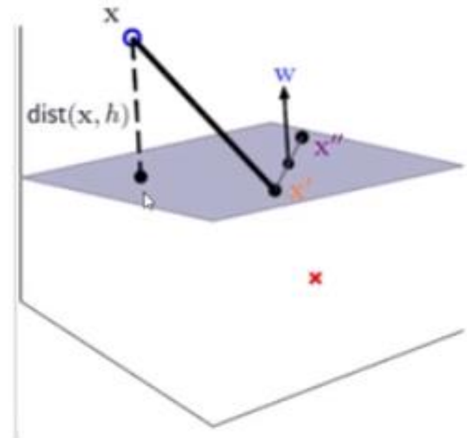
距离的计算

consider $\mathbf{x}'$, $\mathbf{x}''$ on hyperplane

1. $\mathbf{w}^T\mathbf{x}' = -b$, $\mathbf{w}^T\mathbf{x}'' = -b$
2. $\mathbf{w} \perp$ hyperplane:

$$\left(\mathbf{w}^T \underbrace{(\mathbf{x}'' - \mathbf{x}')}_{\text{vector on hyperplane}} \right) = 0$$

3. distance = project $(\mathbf{x} - \mathbf{x}')$ to $\perp$ hyperplane



$$\text{distance}(\mathbf{x}, b, \mathbf{w}) = \left| \frac{\mathbf{w}^T}{\|\mathbf{w}\|}(\mathbf{x} - \mathbf{x}') \right| \overset{①}{=} \frac{1}{\|\mathbf{w}\|}|\mathbf{w}^T\mathbf{x} + b|$$

$$d = \frac{|w^T x + b|}{\|w\|} = \frac{y_i(w^T x_i + b)}{\|w\|}$$

一、线性可分支持向量机

3、最优化问题

$$\arg\max_{w,b}\left\{\frac{1}{\|w\|}\min_{i}\left[y_i\cdot\left(w^T x_i+b\right)\right]\right\}$$

$$y_i(W^T x_i + b) \geq 1$$

$$\arg\max_{w,b}\frac{1}{\|w\|}$$

最优化问题

$$\min\frac{1}{2}w^2$$

$$s.t \quad 1 - y_i(w^T x_i + b) \leq 0 \quad \left(i = 1, 2, \cdots, N\right)$$

一、线性可分支持向量机

4、构造拉格朗日函数

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

$$s.t \qquad \lambda_i \geq 0$$

$$\min_{w,b}\ \max_{\lambda} L(w,b,\lambda)=\frac{1}{2}w^2+\sum_{i=1}^{N}\lambda_i\left(1-y_i\left(w^T x_i+b\right)\right)$$

$$s.t \qquad \lambda_i \geq 0$$

5、强对偶问题

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

一、线性可分支持向量机

6、对w、b求偏导

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{N} \lambda_i y_i x_i = 0$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N} \lambda_i y_i = 0$$

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

一、线性可分支持向量机

7、最终优化问题

$$
\begin{aligned}
\min_{w,b} L(w,b,\lambda) &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j\left(x_i\cdot x_j\right)+\sum_{i=1}^{N}\lambda_i-\sum_{i=1}^{N}\lambda_i y_i\left(\sum_{i=1}^{N}\lambda_i y_i\left(x_i\cdot x_j\right)+b\right)\\
&=\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j\left(x_i\cdot x_j\right)+\sum_{i=1}^{N}\lambda_i-\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j\left(x_i\cdot x_j\right)-\sum_{i=1}^{N}\lambda_i y_i\cdot b\\
&=\sum_{i=1}^{N}\lambda_i-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j\left(x_i\cdot x_j\right)
\end{aligned}
$$

一、线性可分支持向量机

$$\begin{cases}\max\limits_{\lambda}\left[\sum_{i=1}^{N}\lambda_i-\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j\left(x_i\cdot x_j\right)\right] \\ s.t.\quad \sum_{i=1}^{N}\lambda_i y_i=0 \\ \qquad \lambda_i\geq 0\end{cases}$$

$$\begin{cases}\min\limits_{\lambda}\left[\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j\left(x_i\cdot x_j\right)-\sum_{i=1}^{N}\lambda_i\right] \\ s.t.\quad \sum_{i=1}^{N}\lambda_i y_i=0 \\ \qquad \lambda_i\geq 0\end{cases}$$

一、线性可分支持向量机

$$\lambda^{*}=\left(\lambda_{1}^{*},\lambda_{2}^{*},\cdots,\lambda_{N}^{*}\right)$$

$$w^{*}=\sum_{i=1}^{N}\lambda_{i}^{*}y_{i}x_{i} \qquad \left(\text{选择}\lambda_{i}^{*}>0\right)$$

$$b^{*}=y_{j}-\sum_{i=1}^{N}\lambda_{i}^{*}y_{i}\left(x_{i}\cdot x_{j}\right)$$

8、获得超平面

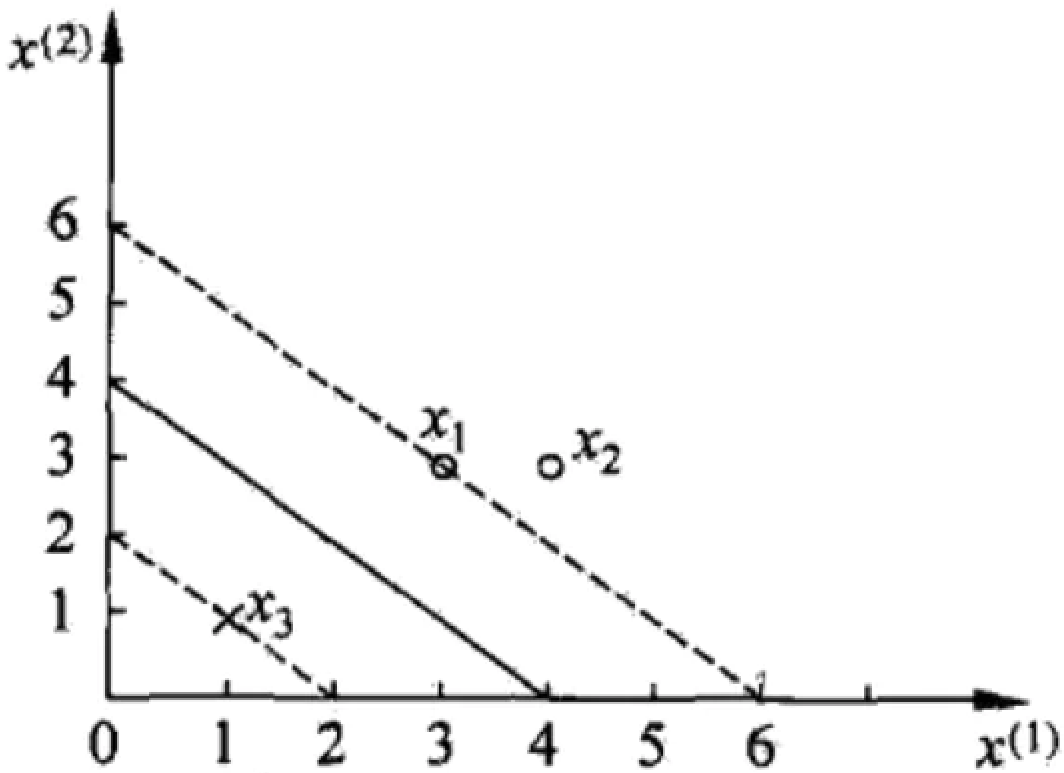
$$w^{*T}x+b^{*}=0$$

9、分类决策

$$f\left(x\right)=\operatorname{sgn}\left(w^{*T}x+b^{*}\right)$$

一、线性可分支持向量机

例：已知一个如图所示的训练数据集，其正例点是$x_1=(3,3)^T$，$x_2=(4,3)^T$，负例点是$x_3=(1,1)^T$，试求最大间隔分离超平面.



一、线性可分支持向量机

$$x_1=(3,3)^T \rightarrow y_1=1$$

$$x_2=(4,3)^T \rightarrow y_2=1$$

$$x_3=(1,1)^T \rightarrow y_3=-1$$

$$\min_{\alpha}\begin{cases}\dfrac{1}{2}\displaystyle\sum_{i=1}^{3}\sum_{j=1}^{3}\alpha_i\alpha_j y_i y_j\left(x_i\cdot x_j\right)-\sum_{i=1}^{3}\alpha_i \\ \displaystyle\sum_{i=1}^{3}\alpha_i y_i=0 \\ \alpha_i\geq 0\end{cases}$$

$$\begin{cases}\dfrac{1}{2}\left(18\alpha_1^2+25\alpha_2^2+2\alpha_3^2+42\alpha_1\alpha_2-12\alpha_1\alpha_3-14\alpha_2\alpha_3\right)-\alpha_1-\alpha_2-\alpha_3 \\ \alpha_1+\alpha_2-\alpha_3=0 \\ \alpha_i\geq 0 \qquad i=1,2,3\end{cases}$$

一、线性可分支持向量机

$$\alpha_3=\alpha_1+\alpha_2$$

$$L=4\alpha_1^2+\frac{13}{2}\alpha_2^2+10\alpha_1\alpha_2-2\alpha_1-2\alpha_2$$

$$\frac{\partial L}{\partial \alpha_1}=0\ ,\quad \frac{\partial L}{\partial \alpha_2}=0\quad 得$$

$\alpha_1=1.5$　　$\alpha_2=-1$　　　不满足 $\alpha_i\geq 0\ ,\ i=1,2,3$

最终的解应该为边界上的点　　$\alpha_1=0,\ \alpha_2=-\frac{2}{13}$

$\alpha_2=0,\ \alpha_1=0.25$

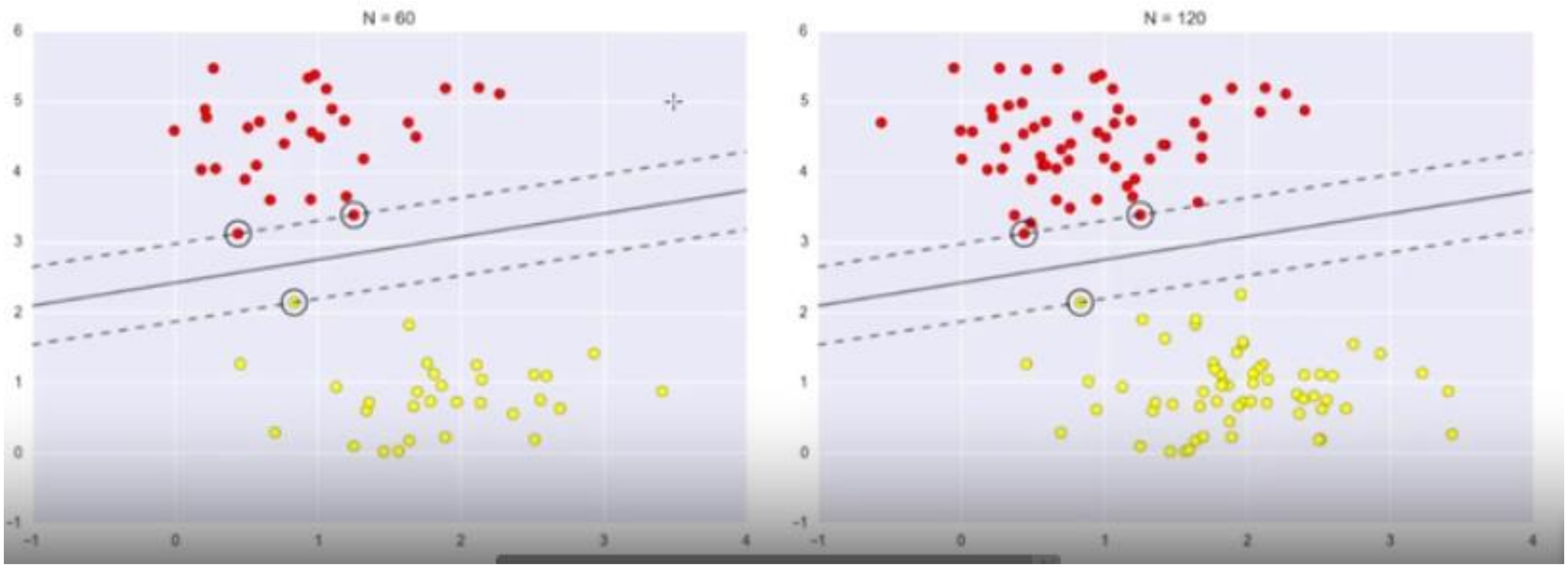
$$\alpha_3=\alpha_1+\alpha_2=0.25$$

最小值在 $(0.25,\ 0\ ,\ 0.25)$　　$w_1=w_2=0.5$　　b = - 2

$$0.5x_1+0.5x_2-2=0$$

一、线性可分支持向量机

支持向量：真正发挥作用的数据点，α值不为0的点



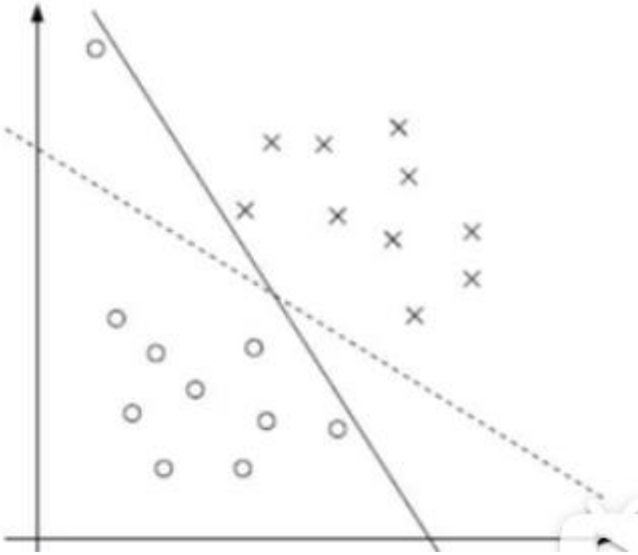
二、软间隔支持向量机

- 软间隔：有时候数据中有一些噪音点，如果考虑它们咱们的线就不太好了

- 之前的方法要求要把两类点完全分得开，这个要求有点过于严格了，我们来放松一点！

- 为了解决该问题，引入松弛因子

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



二、软间隔支持向量机

引入松弛变量 $\xi_i \geq 0$ ，惩罚参数C

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

$$s.t. \qquad y_i(W^T x_i + b) \geq 1 - \xi_i \qquad\qquad i = 1,2,\cdots,N$$
$$\xi_i \geq 0$$

拉格朗日函数

$$\begin{cases} L(w,b,\xi,\alpha,\mu) = \dfrac{1}{2}w^2 + C\sum\limits_{i=1}^{N}\xi_i + \sum\limits_{i=1}^{N}\alpha_i\left(1-\xi_i - y_i\left(w^T x_i + b\right)\right) - \sum\limits_{i=1}^{N}\mu_i\xi_i \\ \alpha_i \geq 0 \\ \mu_i \geq 0 \end{cases}$$

二、软间隔支持向量机

原问题 $$\min_{w,b,\xi}\ \max_{\alpha,\mu} L(w,b,\xi,\alpha,\mu)$$

对偶问题 $$\max_{\alpha,\mu}\ \min_{w,b,\xi}\ L(w,b,\xi,\alpha,\mu)$$

$$\min_{w,b,\xi} L(w,b,\xi,\alpha,\mu)=\min_{w,b,\xi}\left[\frac{1}{2}w^2+C\sum_{i=1}^{N}\xi_i+\sum_{i=1}^{N}\alpha_i\left[1-\xi_i-y_i\left(w^T x_i+b\right)\right]-\sum_{i=1}^{N}\mu_i\xi_i\right]$$

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

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

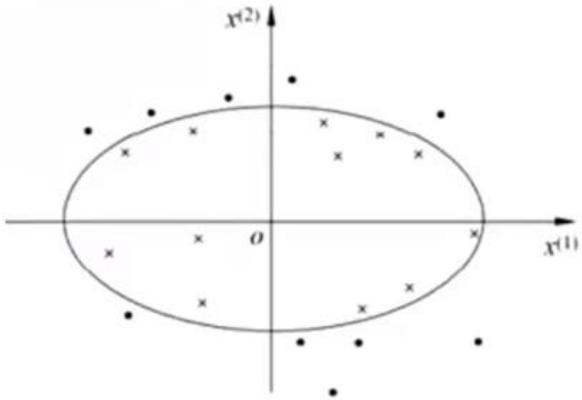
$$\frac{\partial L(w,b,\xi,\alpha,\mu)}{\partial \xi_i}=\mathrm{C}-\alpha_i-\mu_i=0$$

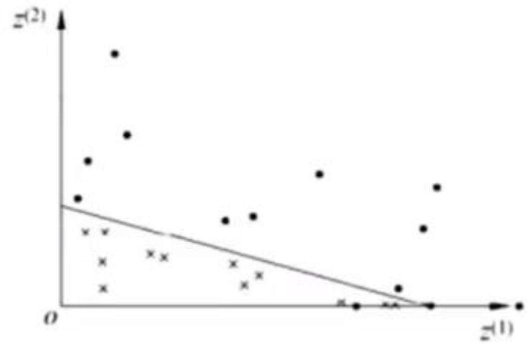
二、软间隔支持向量机

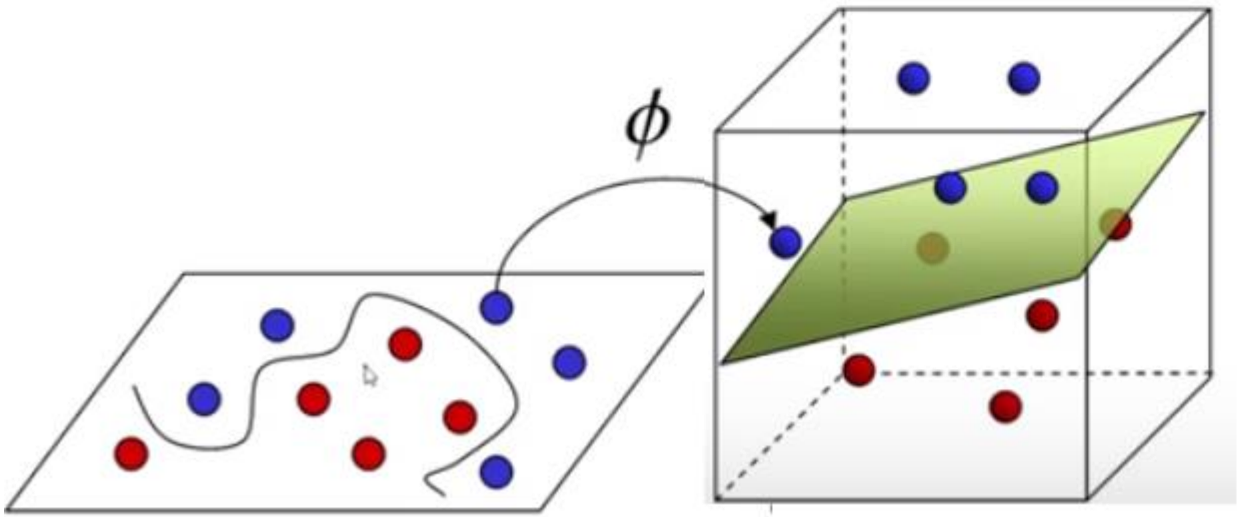
$$
\begin{cases}
w = \sum_{i=1}^{N} \alpha_i y_i x_i \\
\sum_{i=1}^{N} \alpha_i y_i = 0 \\
C\text{-}\alpha_i - \mu_i = 0
\end{cases}
$$

$$
\begin{cases}
\min_{\alpha} L(w,b,\xi,\alpha,\mu) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j \left(x_i \cdot x_j\right) - \sum_{i=1}^{N}\alpha_i \\
s.t. \sum_{i=1}^{N}\alpha_i y_i = 0 \\
0 \le \alpha_i \le C
\end{cases}
$$

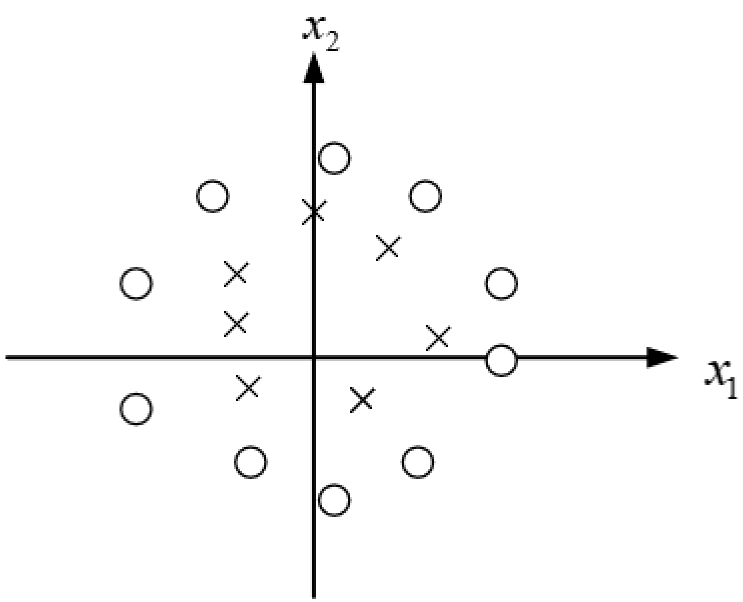
三、非线性支持向量机







三、非线性支持向量机



$$\left(x_1, x_2, x_1^2 + x_2^2\right)$$

$$\left(x_i \cdot x_j\right) = \Phi\left(x_i\right)^T \Phi\left(x_j\right)$$

三、非线性支持向量机

$$x = (x_1, x_2, x_3)$$

$$z = (z_1, z_2, z_3)$$

$$f(x) = (x_1x_1,\ x_1x_2,\ x_1x_3,\ x_2x_1,\ x_2x_2,\ x_2x_3,\ x_3x_1,\ x_3x_2,\ x_3x_3)$$

$$x = (1,2,3) \qquad z = (4,5,6)$$

$$f(x) = (1,2,3,2,4,6,3,6,9)$$

$$f(z) = (16,20,24,20,25,36,24,30,36)$$

$$f(x)^T f(y) = 16 + 40 + 72 + 40 + 100 + 180 + 72 + 180 + 324 = 1024$$

$$(x^T z)^2 = (4 + 10 + 18)^2 = 1024$$

$$K(x,z) = f^T(x) \cdot f(z)$$

三、非线性支持向量机

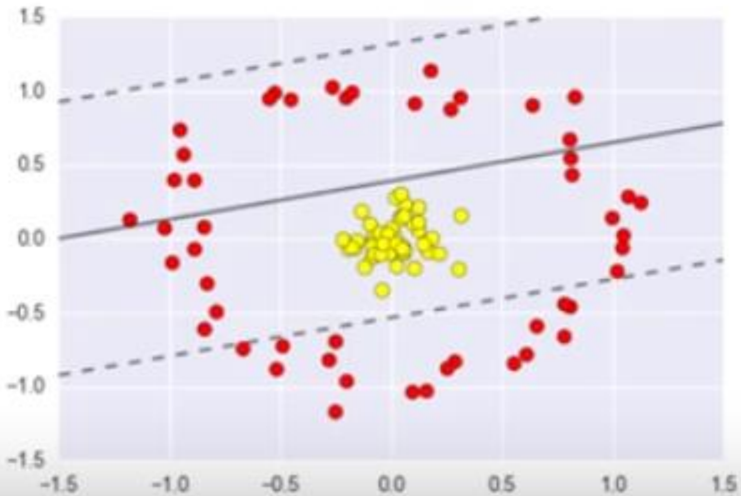
常用核函数

线性核函数： $K(x_i,x_j)=x_i^T x_j$

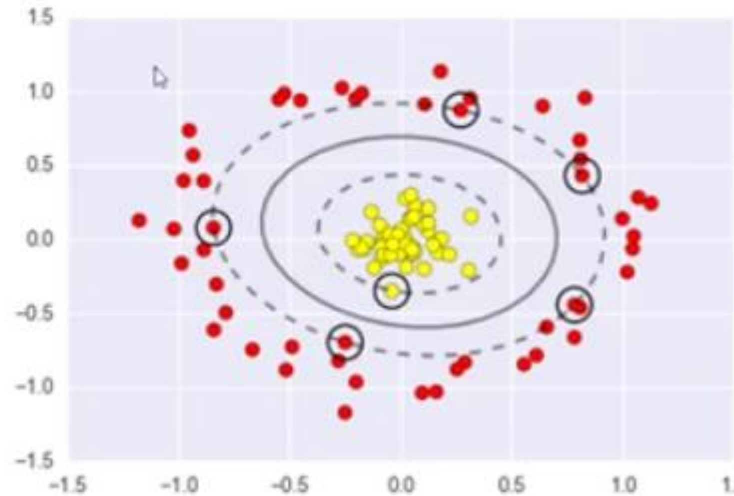
多项式核函数： $K(x_i,x_j)=(x_i^T x_j)^d$

高斯核函数： $K(x_i,x_j)=\exp(-\dfrac{\left\|x_i-x_j\right\|^2}{2\sigma^2})$

三、非线性支持向量机



线性核函数



高斯和函数

四、SMO算法（Sequential Minimal Optimazation)

1、问题

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

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

$$i = 1,2,\ldots,N$$

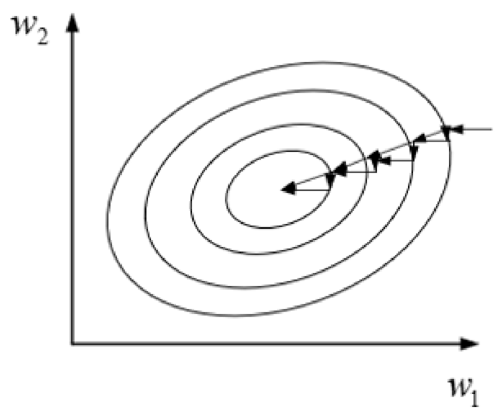
$$0 \le \alpha_i \le C$$

优化变量是拉格朗日乘子 α_i ，一个对应一个样本。

四、SMO算法（Sequential Minimal Optimazation)

2、传统解决方法

（1）梯度下降法



收敛快，计算复杂

（2）坐标下降法

收敛慢，计算简单，（常用方法，但优化 α_i 不能用）

四、SMO算法（Sequential Minimal Optimazation)

3、整体问题化为一系列子问题

每次最少调整两个变量，如果只调整一个，将会破坏 $\sum_{i=1}^{N}\alpha_i y_i = 0$约束条件，只有同时调整两个$\alpha$，才能保证约束条件，这也是$SMO$的由来。这是一种很巧妙的思想。其它不调整的$\alpha$不做任何变化，视为常数，本思路是由John Platt 提出的，这是一种具有影响力的算法，具有与FFT同等地位。

选择两个变量，其它变量固定，SMO将整体优化问题转化成一系列子问题：

四、SMO算法（Sequential Minimal Optimazation)

$$\min_{\alpha_1,\alpha_2} \mathrm{W}(\alpha_1,\alpha_2)=\frac{1}{2}K_{11}\alpha_1^2+\frac{1}{2}K_{22}\alpha_2^2+y_1y_2K_{12}\alpha_1\alpha_2+y_1\alpha_1\sum_{i=3}^{N}y_i\alpha_iK_{i1}+$$

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

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

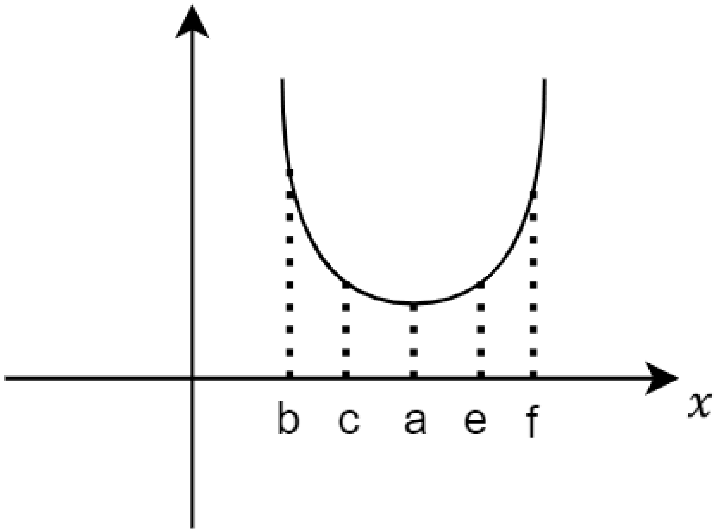
$$0\le\alpha_i\le C, i=1,2$$

$$\begin{cases}\alpha_1y_1=-\alpha_2y_2+\zeta\\ \alpha_1=-\alpha_2y_1y_2+y_1\zeta\end{cases}$$

问题变成了一元函数问题，可以得到解析解，基于初始 可行解 $\alpha_1^{old},\alpha_2^{old}$ ，可以得到 $\alpha_1^{new},\alpha_2^{new}$ ，不用搜索，不用迭代，用解析公式直接求，这就是收敛速度快的原因。

无约束可求得一个解，如果是有约束问题呢？

4、约束与剪辑问题的处理

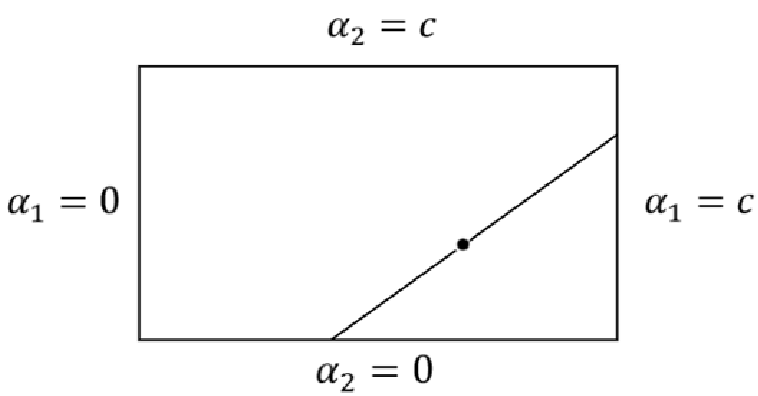


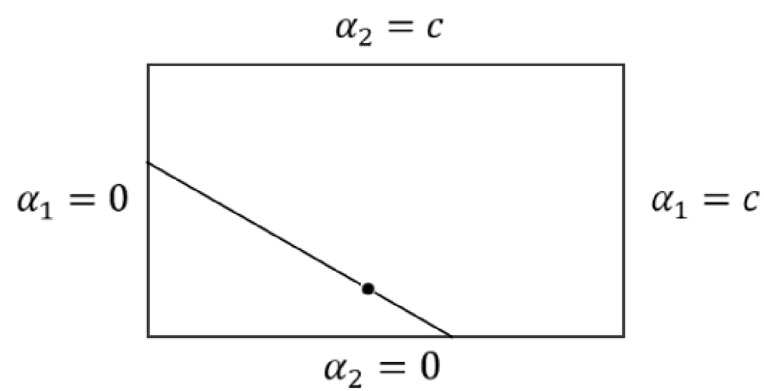
$x\in[b,c]$ $x=c$最小

$x\in[e,f]$ $x=e$最小

$x\in[b,f]$与无约束优化一致

四、SMO算法（Sequential Minimal Optimazation)





$$y_1 \neq y_2 \;\Rightarrow\; \alpha_1 - \alpha_2 = K$$

$$y_1 = y_2 \;\Rightarrow\; \alpha_1 + \alpha_2 = K$$

$$\begin{cases} \alpha_1 y_1 + \alpha_2 y_2 = \zeta \\ \alpha_1 + (-\alpha_2) = y_1 \zeta = K \end{cases}$$

$$y_1 = y_2 \qquad \alpha_1 + \alpha_2 = K$$

四、SMO算法（Sequential Minimal Optimazation)

根据不等式条件α_2^{new}的取值范围（两种情况，$y_1 \neq y_2, y_1 = y_2$ ）

第一种情况 $\alpha_2 = \alpha_1 - K \qquad \alpha_1 \in [0, c]$

$$\alpha_1 - K \in [0 - K, c - K]$$

$$K = \alpha_1 - \alpha_2$$

$$\therefore \alpha_2 \in [\alpha_2 - \alpha_1, c + \alpha_2 - \alpha_1]$$

同时 $\alpha_2 \in [0, c]$

① $$\begin{cases} L \leq \alpha_2^{new} \leq H \\ L = \max\left(0, \alpha_2^{old} - \alpha_1^{old}\right) \\ H = \min\left(c, c + \alpha_2^{old} - \alpha_1^{old}\right) \end{cases}$$

四、SMO算法（Sequential Minimal Optimazation)

第二种情况

$$\alpha_1 \in [0,c]$$

$$K-\alpha_1 \in [K, K-c]$$

$$\alpha_2 \in [K, K-c]$$

同理 $$\begin{cases} \alpha_2 \in [\alpha_1+\alpha_2, \alpha_1+\alpha_2-c] \\ \alpha_2 \in [0,c] \end{cases}$$

② $$\begin{cases} L = \max\left(0, \alpha_2^{old}+\alpha_1^{old}\right) \\ H = \min\left(c, \alpha_1^{old}+\alpha_2^{old}-c\right) \end{cases} \qquad \begin{cases} \alpha_1^{new,\mathrm{un}} \\ \alpha_2^{new,\mathrm{un}} \end{cases}$$ 未剪辑

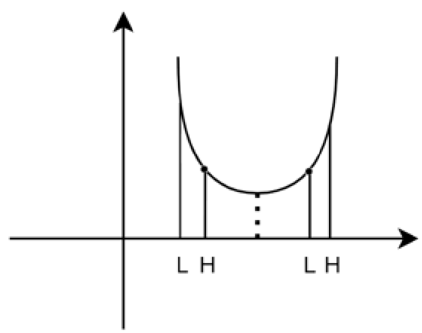
5、求解过程

先求沿着约束方向未经剪辑时 $\alpha_2^{new,un}$ 再求剪辑后的 α_2^{new}

$$\text{记：}\quad g(x)=\sum^{N}\alpha_i y_i K\left(x_i,x\right)+b$$

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

E为输入 x 的预测值和真实输出 y的差，$i=1,2$



引进记号

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

四、SMO算法（Sequential Minimal Optimazation)

目标函数写成

$$W(\alpha_1,\alpha_2)=\frac{1}{2}K_{11}\alpha_1^2+\frac{1}{2}K_{22}\alpha_2^2+y_1y_2K_{12}\alpha_1\alpha_2-(\alpha_1+\alpha_2)+y_1v_1\alpha_1+y_2v_2\alpha_2$$

由 $\alpha_1 y_1=\zeta-\alpha_2 y_2$　　设 $y_i^2=1$

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

得到只有 α_2 的目标函数

$$W(\alpha_2)=\frac{1}{2}K_{11}(\zeta-\alpha_2y_2)^2+\frac{1}{2}K_{22}\alpha_2^2+y_2K_{12}(\zeta-\alpha_2y_2)\alpha_2$$
$$-(\zeta-\alpha_2y_2)y_1-\alpha_2+v_1(\zeta-\alpha_2y_2)+y_2v_2\alpha_2$$

四、SMO算法（Sequential Minimal Optimazation)

对α_2求导

$$\frac{\partial W}{\partial \alpha_2}=K_{11}\alpha_2+K_{22}\alpha_2-2K_{12}\alpha_2-K_{11}\zeta y_2+K_{12}\zeta y_2+y_1y_2-1-v_1y_2+y_2v_2=0$$

$$\left(K_{11}+K_{22}-2K_{12}\right)\alpha_2=y_2\left(y_2-y_1+\zeta K_{11}-\zeta K_{12}+v_1-v_2\right)$$

$$=y_2\left(y_2-y_1+\zeta K_{11}-\zeta K_{12}+\left(g(x_1)-\sum_{j=1}^{2}y_j\alpha_jK_{1j}-b\right)-\left(g(x_2)-\sum_{j=1}^{2}y_j\alpha_jK_{2j}-b\right)\right)$$

式子虽长，但只有一个变量，最高二次方

将$\zeta=\alpha_1^{old}y_1+\alpha_2^{old}y_2$代入

$$\left(K_{11}+K_{22}-2K_{12}\right)\alpha_2^{\text{new,un}}=y_2\left(\left(K_{11}+K_{22}-2K_{12}\right)\alpha_2^{old}y_2+y_2-y_1+g\left(x_1\right)-g\left(x_2\right)\right)$$

$$=\left(K_{11}+K_{22}-2K_{12}\right)\alpha_2^{old}+y_2\left(E_1-E_2\right)$$

四、SMO算法（Sequential Minimal Optimazation)

令 $\eta = K_{11} + K_{22} - 2K_{12}$

$$\alpha_2^{\text{new,un}} = \alpha_2^{old} + \frac{y_2\left(E_1 - E_2\right)}{\eta}$$

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

剪辑后得

$$\alpha_2^{new} = \begin{cases} H & \alpha_2^{new,un} > H \\ \alpha_2^{new,un} & \mathrm{L} \le \alpha_2^{new,un} \le H \\ L & \alpha_2^{new,un} < \mathrm{L} \end{cases}$$

四、SMO算法（Sequential Minimal Optimazation）

得到 α_1 的解

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

为了下一轮迭代，需要计算 b 和 E_i 完成两个变量的优化后，

重新计算 b 和E_i 由KKT条件，如果 $0 < \alpha_1^{new} < c$ （支持向量）

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

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

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

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

四、SMO算法（Sequential Minimal Optimazation)

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

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

b_1^{new}只与α_1、α_2有关

如果 $0< \alpha_2^{new} < c$

$$b_2^{new} = -E_2 - y_1 K_{12}\left(\alpha_1^{new} - \alpha_1^{old}\right) - y_2 K_{22}\left(\alpha_2^{new} - \alpha_2^{old}\right) + b^{old}$$

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

S是所有支持向量 x_j 的集合

如果$\alpha_1^{new},\alpha_2^{new}$同时满足条件$0<\alpha_i^{new}<c\left(i=1,2\right)$,那么 $b_1^{new}=b_2^{new}$。如果 $\alpha_1^{new},\alpha_2^{new}$是 0或者c,那么$b_1^{new}$和$b_2^{new}$以及它们之间的数都是符合 KKT条件的阈值，这时选择它们的中点作为b ^{new}。

以上是SMO的主要框架

6、变量的启发式选择

SMO算法在每个子问题中选择两个变量，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

$$\alpha_i = 0 \leftrightarrow y_i g(x_i) > 1$$

$$0 < \alpha_i < c \leftrightarrow y_i g(x_i) = 1$$

$$\alpha_i = c \leftrightarrow y_i g(x_i) < 1$$

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

四、SMO算法（Sequential Minimal Optimazation)

2、第二个变量的检查（内循环）

选择的标准量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$\mathbf{E_1}$，$\mathbf{E_2}$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大的下降，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。

SMO算法流程略。