贪心科技
让每个人享受个性化教育服务

KKT Condition

Primal Dual(1)

- 对于原优化问题:

$$\min_{w} f(w)$$
$$s.t. \quad \begin{matrix} g_i(w) \le 0 \\ h_i(w) = 0 \end{matrix}$$

- 其拉格朗日函数如下:

$$\mathcal{L}(w, \alpha, \beta) = f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w)$$

- 其中α, β为拉格朗日算子.
- 定义

$$\theta_{\mathrm{p}}(w) = \max_{\alpha, \beta : \alpha_i \ge 0} \mathcal{L}(w, \alpha, \beta)$$

对于给定的w, 如果原问题的约束没有被满足(例如对于某个 i: $g_i(w) > 0$或者 $h_i(w) \ne 0$), 可以得知

$$\theta_{\mathrm{p}}(w) = \max_{\alpha, \beta : \alpha_i \ge 0} \left[f(w) + \sum_{i=1}^{k} \alpha_i g_i(w) + \sum_{i=1}^{l} \beta_i h_i(w) \right] = \infty$$

相反, 如果对于给定的w, primal的约束被满足, 那么$\theta_{\mathrm{p}}(w) = f(w)$, 即

$$\theta_{\mathrm{p}}(w) = \begin{cases} f(w) & \text{约束被满足} \\ \infty & \text{约束没有被满足} \end{cases}$$

Primal Dual(2)

如下最小化 primal 问题与原问题具有同样的解

$$\min_{w} \theta_{\mathfrak{p}}(w) = \min_{w} \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta)$$

定义 $\theta_{\mathcal{D}}(\alpha,\beta) = \min_{w} \mathcal{L}(w,\alpha,\beta)$, 注意$\theta_{\mathfrak{p}}$是针对$\alpha,\beta$的优化(求最大), $\theta_{\mathcal{D}}(\alpha,\beta)$是针对$w$优化(最小化). 我们可以将 dual 优化问题定义为

$$\max_{\alpha,\beta:\alpha_i \geq 0} \theta_{\mathcal{D}}(\alpha,\beta) = \max_{\alpha,\beta:\alpha_i \geq 0} \min_{w} \mathcal{L}(w,\alpha,\beta)$$

我们可以定义$p^* = \min_{w} \theta_{\mathfrak{p}}(w)$ 为 primal 最小化问题的值, $d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \theta_{\mathcal{D}}(\alpha,\beta)$ 为dual 问题的值. 他们的关系如下(一个函数的 “max min” 总是小于等于 “min max”):

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

Primal Dual(3): KKT

已知 $d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \min_w \mathcal{L}(w,\alpha,\beta) \leq \min_w \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta) = p^*$

只有在KKT条件下: $d^* = p^*$, 在这种条件下, 我们就可以通过求解 dual 问题来求解 primal 问题.

假设 f, g_i 都是 convex 函数($f(w) = w^T w$是 convex的,g_i是affine的, affine函数都是convex的), h_i 都是 affine 的($h_i(w) = a_i^T w + b$), 同时假设存在w使得$g_i(w) < 0$对所有 i 都成立, 那么一定存在w^*, α^*, β^*满足Karush-Kuhn-Tucker(KKT)条件, 同时w^*, α^*, β^*也是 primal 和 dual 问题的解. KKT 条件如下:

$$
\begin{aligned}
\frac{\partial}{\partial w_i}\mathcal{L}(w^*,\alpha^*,\beta^*) &= 0, \quad i = 1,\ldots,d \\
\frac{\partial}{\partial \beta_i}\mathcal{L}(w^*,\alpha^*,\beta^*) &= 0, \quad i = 1,\ldots,l \\
\alpha_i^* g_i(w^*) &= 0, \quad i = 1,\ldots,k \\
g_i(w^*) &\leq 0, \quad i = 1,\ldots,k \\
\alpha^* &\geq 0, \quad i = 1,\ldots,k
\end{aligned}
$$

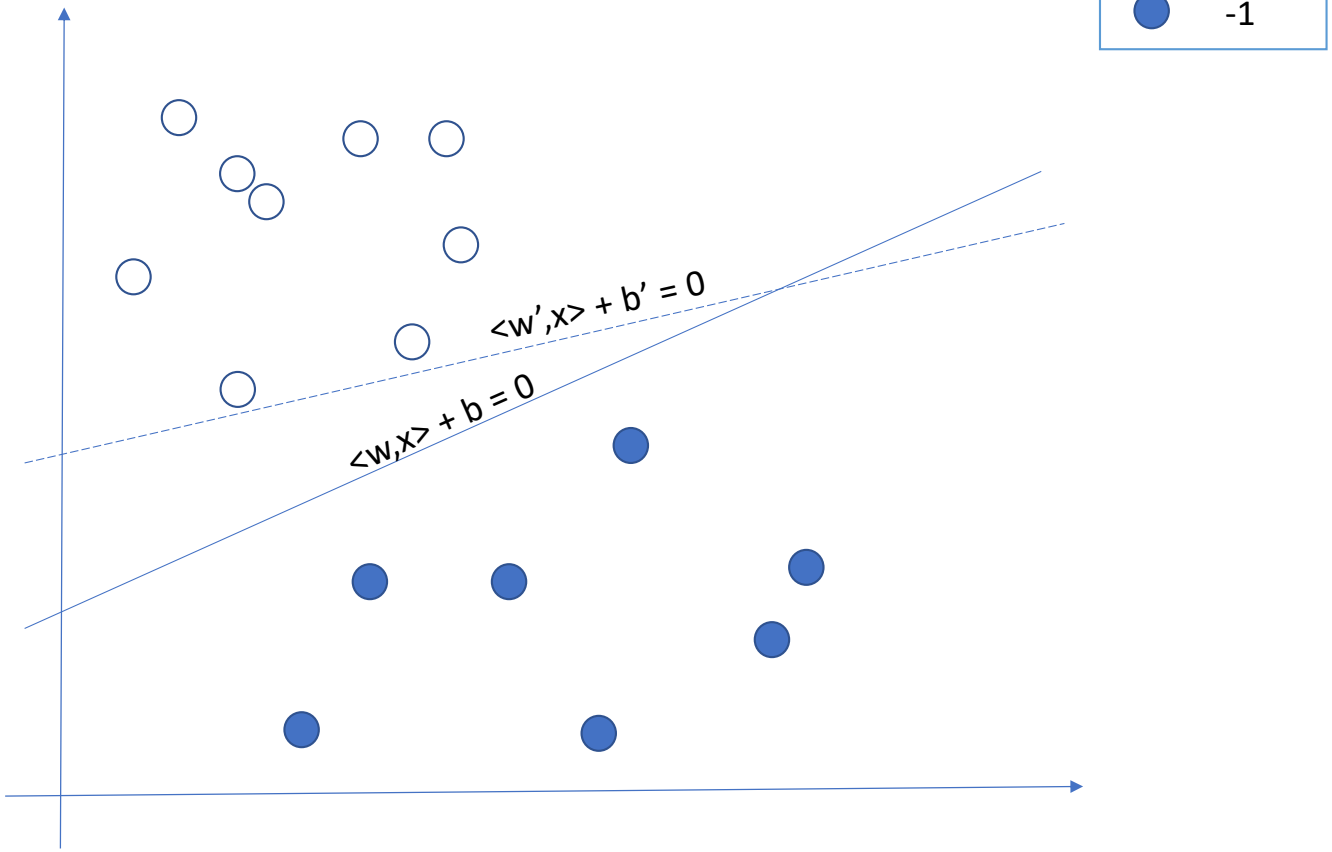
dual complementarity, 当$\alpha_i^* > 0, g_i(w^*) = 0$

- 证明: $d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \min_{w} \mathcal{L}(w,\alpha,\beta) \leq \min_{w} \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta) = p^*$
- 对任意(w,α,β), 如下不等式一定成立:
  - $\theta_{\mathcal{D}}(\alpha,\beta) = \min_{w} \mathcal{L}(w,\alpha,\beta) \leq \mathcal{L}(w,\alpha,\beta) \leq \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta) = \theta_{\mathfrak{p}}(w)$
- 即:
  - $\theta_{\mathcal{D}}(\alpha,\beta) \leq \theta_{\mathfrak{p}}(w)$,
- 所以:
  - $\max_{\alpha,\beta:\alpha_i \geq 0} \theta_{\mathcal{D}}(\alpha,\beta) \leq \min_{w} \theta_{\mathfrak{p}}(w)$
- 即得到:
  - $d^* = \max_{\alpha,\beta:\alpha_i \geq 0} \min_{w} \mathcal{L}(w,\alpha,\beta) \leq \min_{w} \max_{\alpha,\beta:\alpha_i \geq 0} \mathcal{L}(w,\alpha,\beta) = p^*$

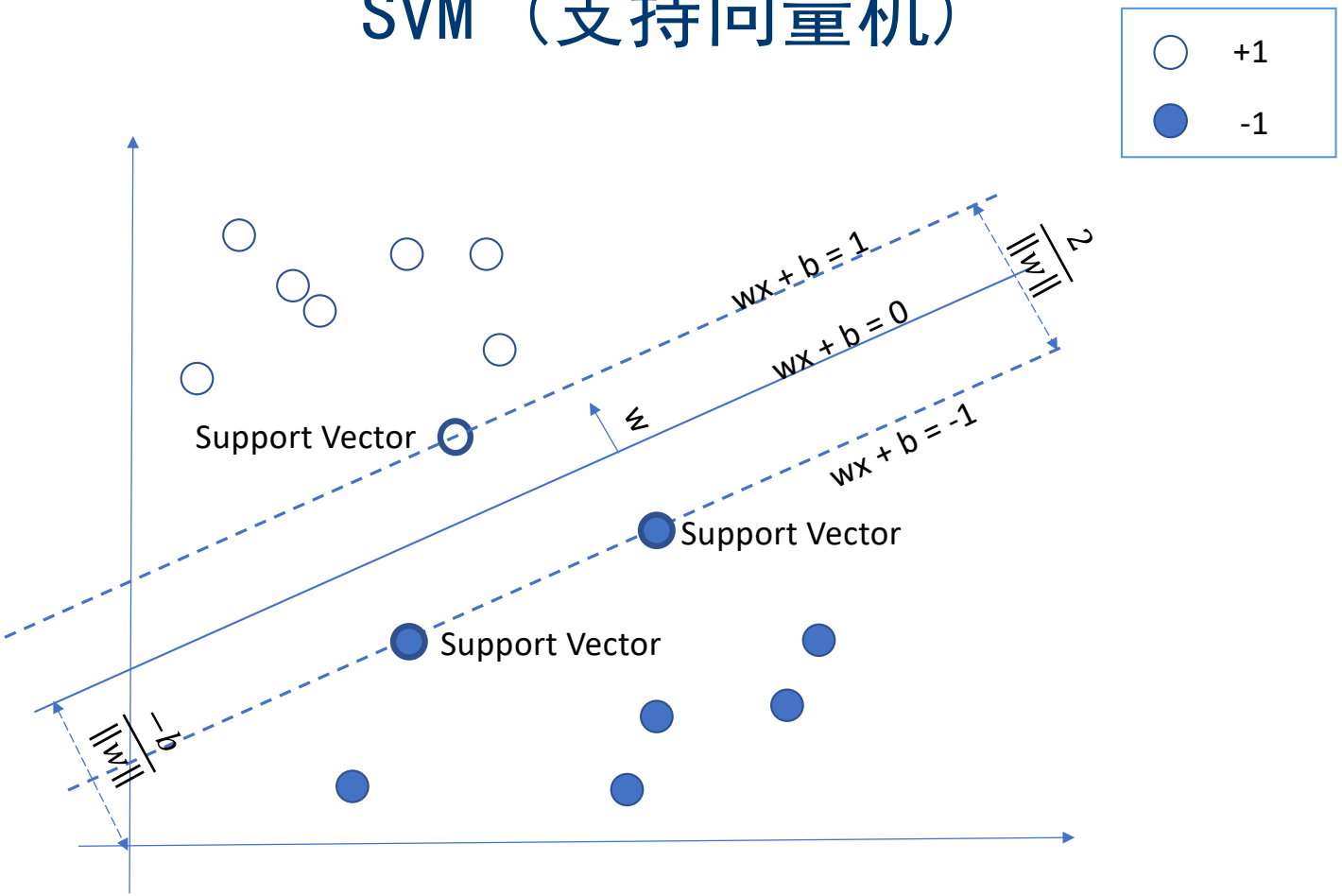
贪心科技
让每个人享受个性化教育服务

支持向量机SVM

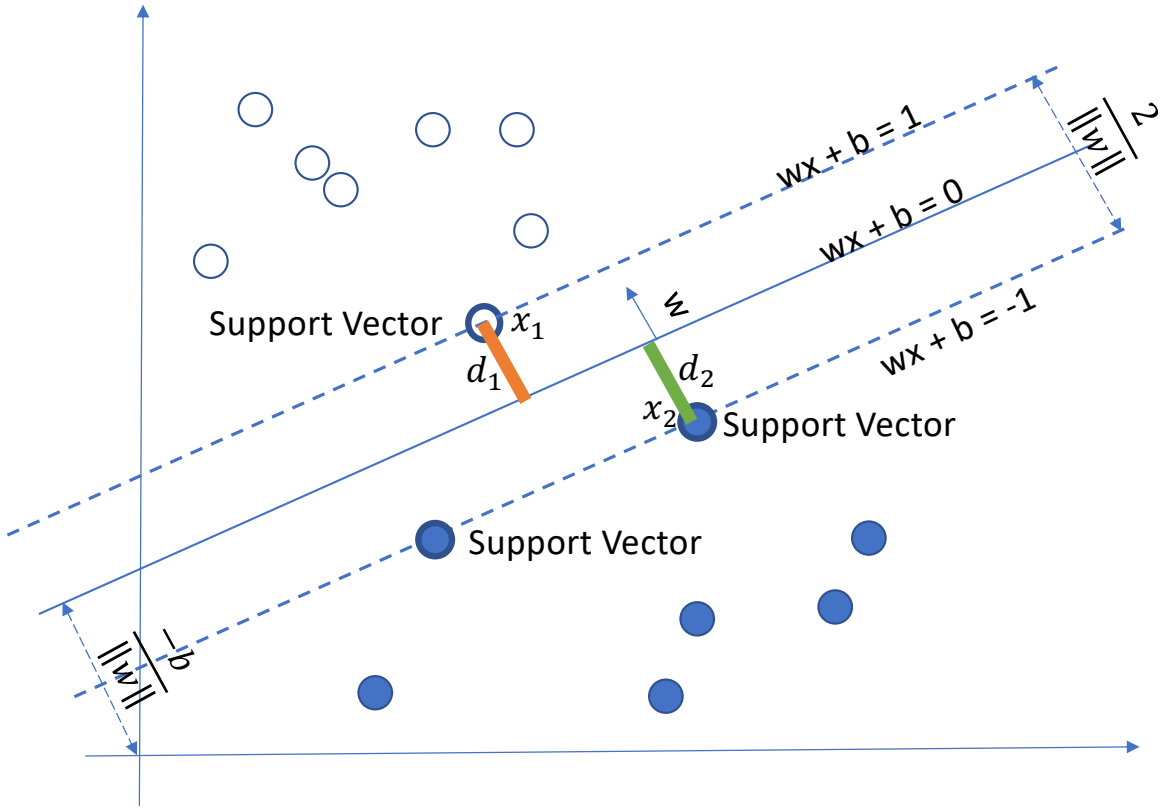
线性分类器



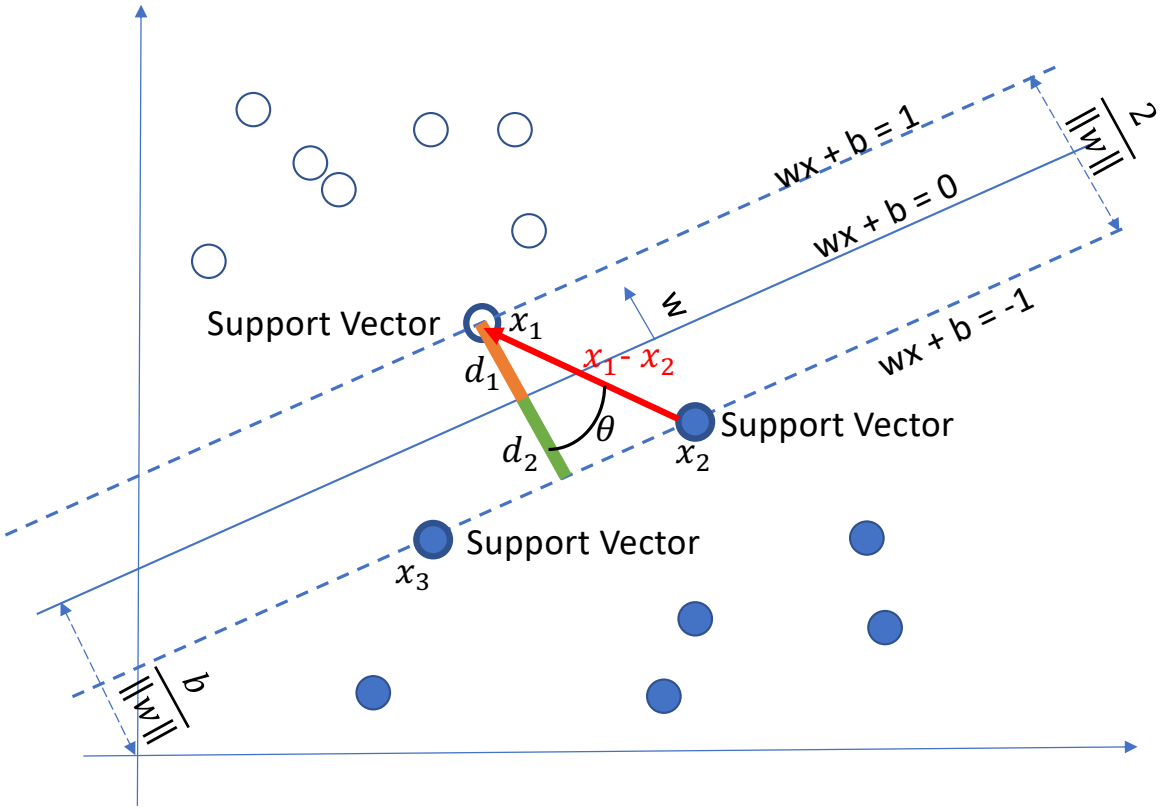
SVM（支持向量机）


+1
-1
wx + b = 1
wx + b = 0
wx + b = -1
Support Vector
Support Vector
Support Vector
w
2/||w||
-b/||w||

SVM（支持向量机）


+1
-1
wx + b = 1
wx + b = 0
wx + b = -1
2/||w||
-b/||w||
w
Support Vector
Support Vector
Support Vector
x_1
x_2
d_1
d_2

SVM（支持向量机，Max-Margin Classifier）



$$w^T x_1 + b = 1$$

$$w^T x_2 + b = -1$$

$$(w^T x_1 + b) - (w^T x_2 + b) = 2$$

$$w^T (x_1 - x_2) = 2$$

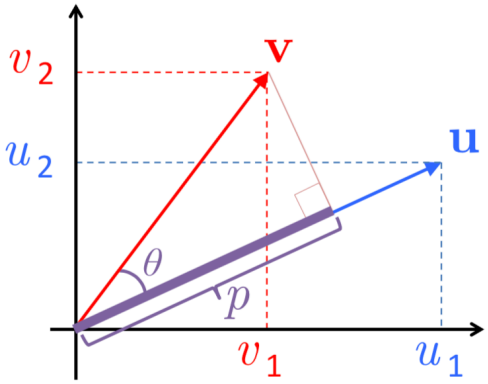
$$w^T (x_1 - x_2) = \|w\|_2 \|x_1 - x_2\|_2 cos\theta = 2$$

$$\|x_1 - x_2\|_2 cos\theta = \frac{2}{\|w\|_2}$$

$$d_1 = d_2 = \frac{\|x_1 - x_2\|_2 cos\theta}{2} = \frac{\frac{2}{\|w\|_2}}{2} = \frac{1}{\|w\|_2}$$

$$d_1 + d_2 = \frac{2}{\|w\|_2}$$

向量内积



$$u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \qquad v = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

$$\|\mathbf{u}\|_2 = \text{length}(\mathbf{u}) \in \mathbb{R}$$
$$= \sqrt{u_1^2 + u_2^2}$$

$$\mathbf{u}^\intercal \mathbf{v} = \mathbf{v}^\intercal \mathbf{u}$$
$$= u_1 v_1 + u_2 v_2$$
$$= \|\mathbf{u}\|_2 \ \|\mathbf{v}\|_2 \cos\theta$$
$$= p\|\mathbf{u}\|_2 \quad \text{where } p = \|\mathbf{v}\|_2 \cos\theta$$

SVM 数学模型（1）

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

$$\text{s.t.} \quad y^{(i)}(w^T x^{(i)} + b) \geq 1, \quad i = 1, \ldots, n$$

可以将约束条件写为:

$$g_i(w) = -y^{(i)}(w^T x^{(i)} + b) + 1 \leq 0$$

可以将优化问题拉格朗日化:

$$\mathcal{L}(w, b, \alpha) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} \alpha_i \left[y^{(i)}(w^T x^{(i)} + b) - 1 \right]$$

SVM 数学模型 (2)

$$\mathcal{L}(w,b,\alpha)=\frac{1}{2}||w||^2-\sum_{i=1}^{n}\alpha_i\left[y^{(i)}(w^Tx^{(i)}+b)-1\right]$$

欲构造 dual 问题, 首先求拉格朗日化的问题中 w 和 b 的值, 对 w 求梯度, 令梯度为 0, 可求得 w:

$$\nabla_w\mathcal{L}(w,b,\alpha)=w-\sum_{i=1}^{n}\alpha_iy^{(i)}x^{(i)}=0 \quad\Longrightarrow\quad w=\sum_{i=1}^{n}\alpha_iy^{(i)}x^{(i)}$$

对 b 求梯度, 令梯度为 0, 可得 :

$$\frac{\partial}{\partial b}\mathcal{L}(w,b,\alpha)=\sum_{i=1}^{n}\alpha_iy^{(i)}=0$$

将 w 带入拉格朗日化的原问题可得 :

$$\mathcal{L}(w,b,\alpha)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}y^{(i)}y^{(j)}\alpha_i\alpha_j(x^{(i)})^Tx^{(j)}-b\sum_{i=1}^{n}\alpha_iy^{(i)}$$

$$\mathcal{L}(w,b,\alpha)=\sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}y^{(i)}y^{(j)}\alpha_i\alpha_j(x^{(i)})^Tx^{(j)}$$

SVM 数学模型 (3)

对拉格朗日化的原问题求最小值, 得到了 w, 现在可以构造 dual 问题 :

$$\max_\alpha \quad W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle$$

$$\text{s.t.} \quad \alpha_i \geq 0, \quad i = 1, \ldots, n$$

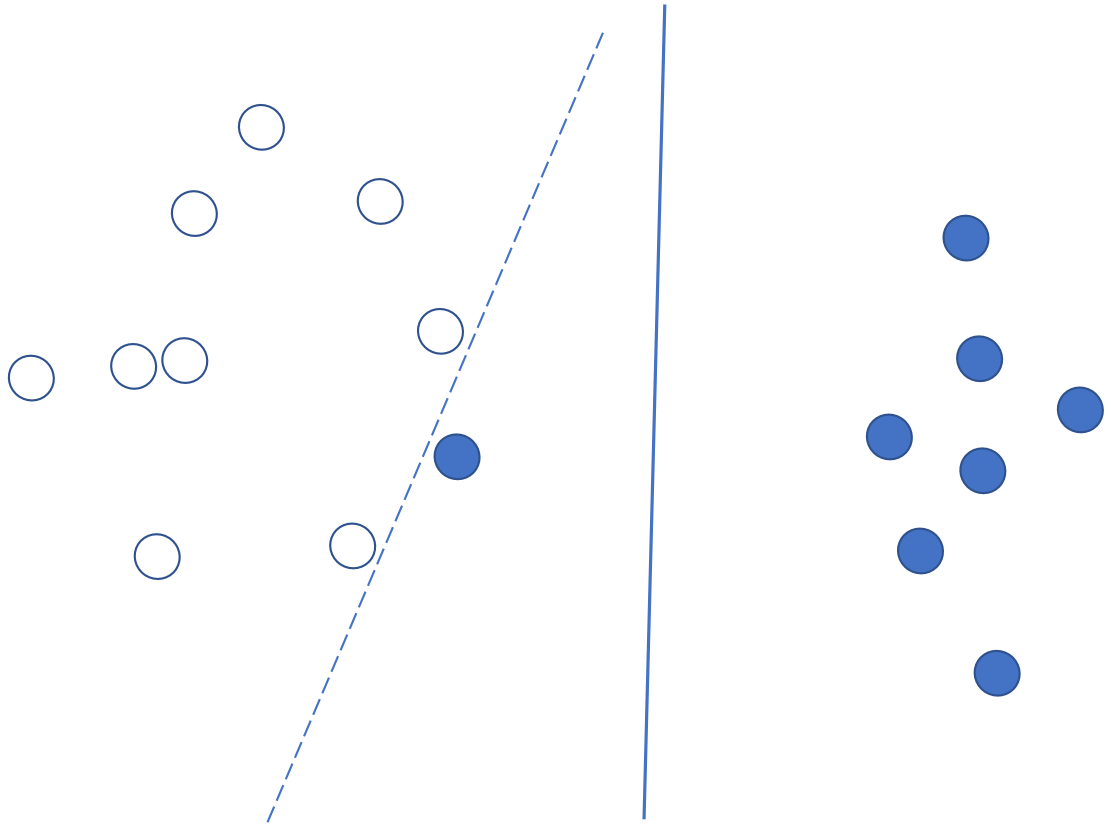
$$\sum_{i=1}^{n} \alpha_i y^{(i)} = 0,$$

可以推导出 b 的值为(作业): $b^* = -\frac{\max_{i:y^{(i)}=-1} w^{*T} x^{(i)} + \min_{i:y^{(i)}=1} w^{*T} x^{(i)}}{2}$

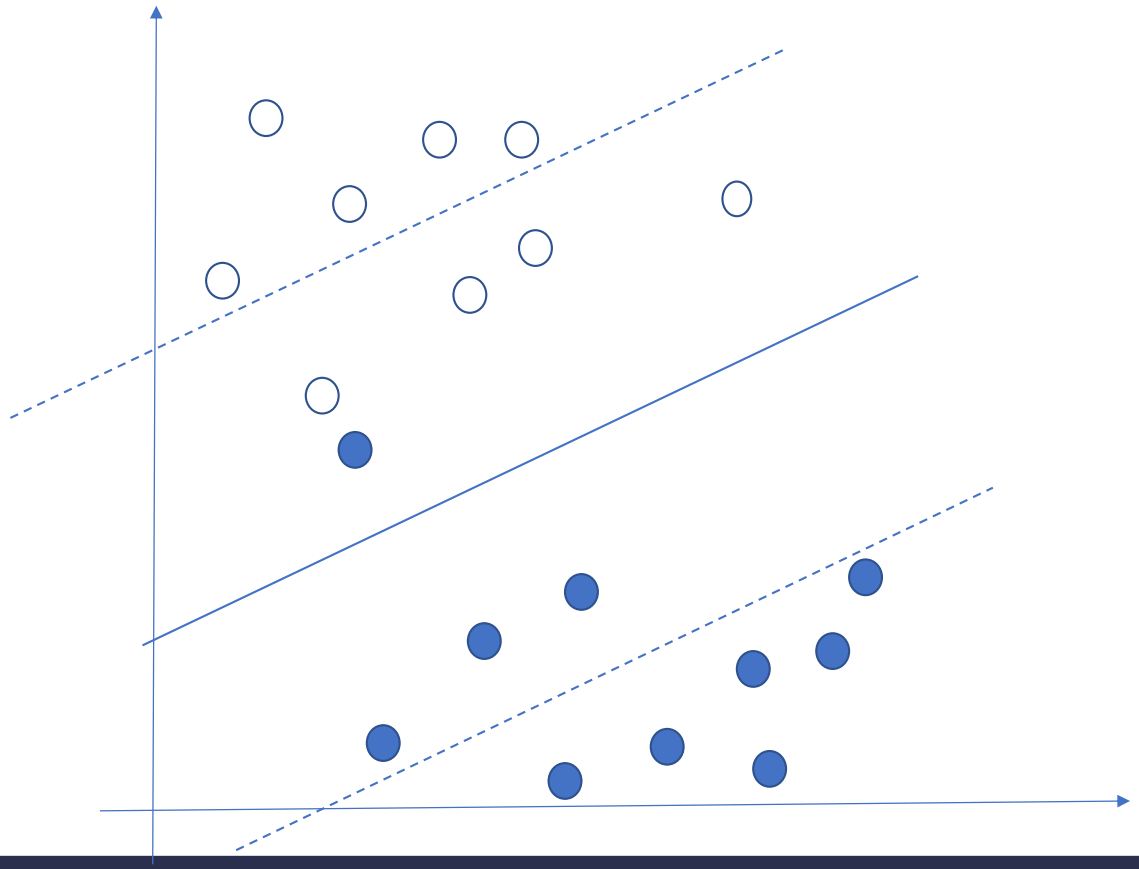
SVM的决策子如下, 值的符号为类别.

$$w^T x + b = \left(\sum_{i=1}^{n} \alpha_i y^{(i)} x^{(i)} \right)^T x + b = \sum_{i=1}^{n} \alpha_i y^{(i)} \langle x^{(i)}, x \rangle + b$$

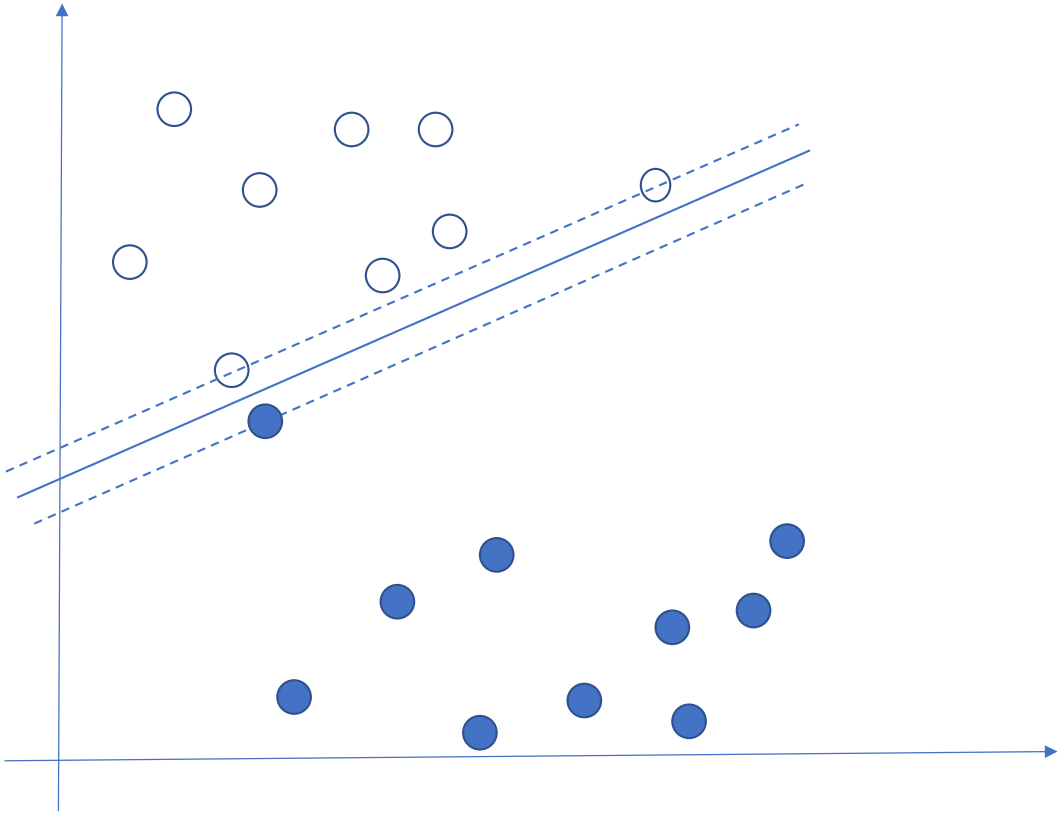
SVM 异常值（outlier）



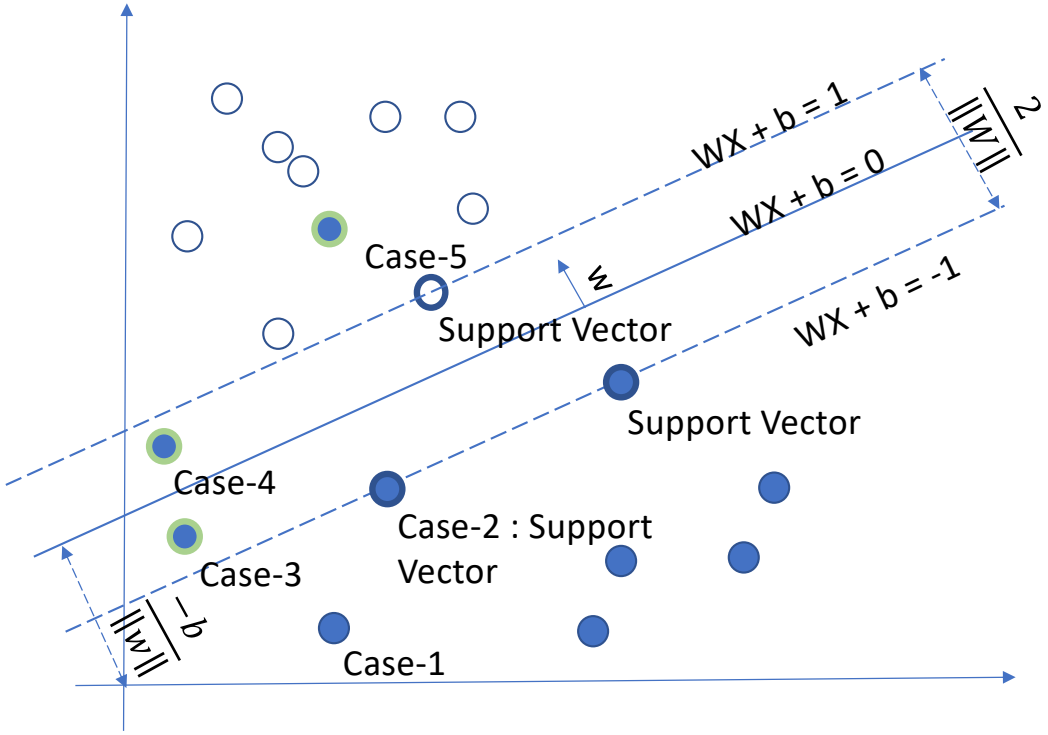
SVM 异常值（outlier）处理一：放松限制



SVM 异常值（outlier）处理二：不放松限制



SVM 异常值（outlier）处理三 ： 必须放松限制（处理线性不可分的情形）



带松弛变量的 SVM 数学模型（1）

原问题:

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

$$\text{s.t.} \quad y^{(i)}(w^T x^{(i)} + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n$$

$$\xi_i \geq 0, \quad i = 1, \ldots, n.$$

Dual 问题:

$$\max_{\alpha} \quad W(\alpha) = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} y^{(i)}y^{(j)}\alpha_i\alpha_j\langle x^{(i)}, x^{(j)}\rangle$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq C, \quad i = 1, \ldots, n$$

$$\sum_{i=1}^{n}\alpha_i y^{(i)} = 0,$$

带松弛变量的 SVM 数学模型（2）

KKT dual-complementarity 条件:

$$\begin{aligned} \alpha_i = 0 &\Rightarrow y^{(i)}(w^T x^{(i)} + b) \geq 1 \\ \alpha_i = C &\Rightarrow y^{(i)}(w^T x^{(i)} + b) \leq 1 \\ 0 < \alpha_i < C &\Rightarrow y^{(i)}(w^T x^{(i)} + b) = 1 \end{aligned}$$

Hinge Loss（合页损失函数）

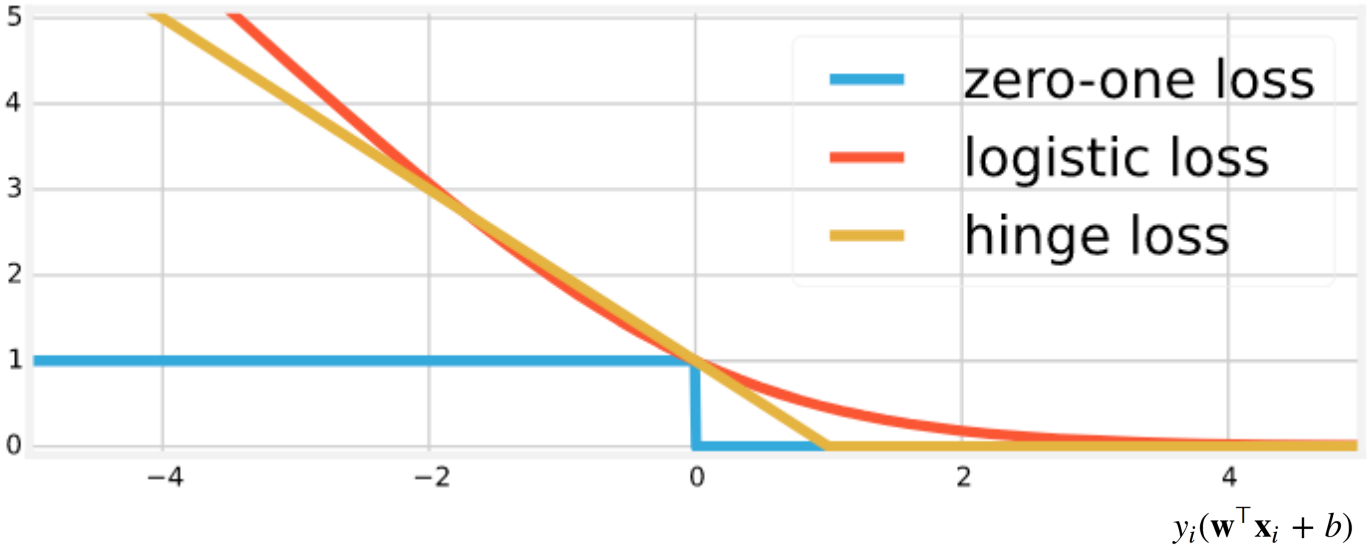
$$y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i \Rightarrow \xi_i \geq 1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b)$$

$$\xi_i \geq 0$$

$$\xi_i = \max(0, 1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b))$$

Hinge Loss（合页损失函数）

$$\xi_i = \max(0, 1 - y_i(\mathbf{w}^\top \mathbf{x}_i + b))$$



Hinge Loss（合页损失函数）

- Convex 凸函数, 容易优化
- 在自变量小于0的部分梯度比较小, 对错误分类的惩罚比较轻
- 在自变量大于等于1的部分, 值为0: 只要对某个数据分类是正确的, 并且正确的可能性足够高, 那么就用不着针对这个数据进一步优化了
- 在自变量等于0处不可导, 需要分段求导
- 使得在求解最优化时, 只有支持向量(support vector)会参与确定分界线, 而且支持向量的个数远小于训练数据的个数

SVM 扩展内容

带松弛变量的 SVM 数学模型

$$\min_{\mathbf{w},b,\xi\geq 0} \quad \frac{1}{2}\mathbf{w}^{\top}\mathbf{w} + C\sum_{i}\xi_i$$

$$\text{s.t.} \quad y_i(\mathbf{w}^{\top}\mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, \ldots, n$$

$$\xi_i \geq 0$$

$$h(\mathbf{x}) = sign(\mathbf{w}^{\top}\mathbf{x} + b)$$

将原 SVM 最优化问题，添加拉格朗日算子，转化为一个新的最优化问题

$$L(\mathbf{w}, b, \xi, \alpha, \lambda) = \frac{1}{2}\mathbf{w}^\top\mathbf{w} + C\sum_i \xi_i + \sum_i \alpha_i(1 - \xi_i - y_i(\mathbf{w}^\top\mathbf{x}_i + b)) - \sum_i \lambda_i\xi_i$$

$$\frac{\partial L}{\partial b} = 0 \quad \rightarrow \quad \sum_i \alpha_i y_i = 0$$

$$\frac{\partial L}{\partial \mathbf{w}} = 0 \quad \rightarrow \quad \mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$$

$$\frac{\partial L}{\partial \xi_i} = 0 \quad \rightarrow \quad C - \alpha_i - \lambda_i = 0$$

将原 SVM 最优化问题，添加拉格朗日算子，转化为一个新的最优化问题（续）

代入 $\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i$ $\sum_i \alpha_i y_i = 0$

$$L(\xi, \alpha, \lambda) = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j + \sum_i \xi_i (C - \alpha_i - \lambda_i)$$

代入 $C - \alpha_i - \lambda_i = 0$

$$\max_{\alpha \geq 0, \lambda \geq 0} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j$$

$$\text{s.t.} \sum_i \alpha_i y_i = 0, \quad C - \alpha_i - \lambda_i = 0$$

将原 SVM 最优化问题，添加拉格朗日算子，转化为一个新的最优化问题（续）

$$\max_{\alpha \geq 0, \lambda \geq 0} \quad \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j$$

$$\text{s.t.} \quad \sum_i \alpha_i y_i = 0, \quad C - \alpha_i - \lambda_i = 0$$

由于λ_i唯一需要满足的条件是大于等于 0，

约束条件 $C - \alpha_i - \lambda_i = 0$ 也可以改为：

$$\alpha_i \leq C$$

SVM 对偶形式

两个数据点属于同一类别使值增加，否则减小

衡量两个数据之间的相似性

$$\max_{\alpha \geq 0, \lambda \geq 0} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^\top \mathbf{x}_j$$

$$\text{s.t.} \quad \sum_i \alpha_i y_i = 0, \quad C - \alpha_i - \lambda_i = 0$$

不同数据点的权重不同，不同的类别的权重一致

使用核函数

$$\max_{\alpha \geq 0} \quad \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$$

$$\text{s.t.} \quad \sum_i \alpha_i y_i = 0, \quad \alpha_i \leq C, \quad i = 1, \ldots, n$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$$

使用核函数 $k(\mathbf{x}_i, \mathbf{x}_j)$ 替代 $\mathbf{x}_i^\top \mathbf{x}_j$

The Kernel Trick（核技巧）

- 如果一个算法可以表达为关于一个正定核 K1 的函数, 那么可以将它转化为关于另外一个正定核 K2 的函数

- SVM 可以使用 The Kernel Trick

使用核函数

两个数据点属于同一类别使值增加，否则减小

衡量两个数据之间的相似性

$$\max_{\alpha \geq 0} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$$

$$\text{s.t.} \quad \sum_i \alpha_i y_i = 0, \quad \alpha_i \leq C, \quad i = 1, \ldots, n$$

不同数据点的权重不同，不同的类别的权重一致

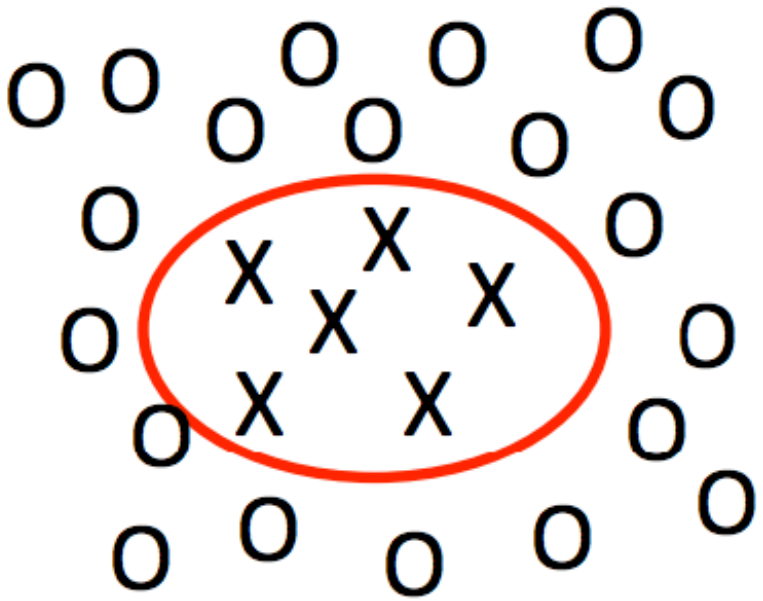
使用核函数, 预测公式

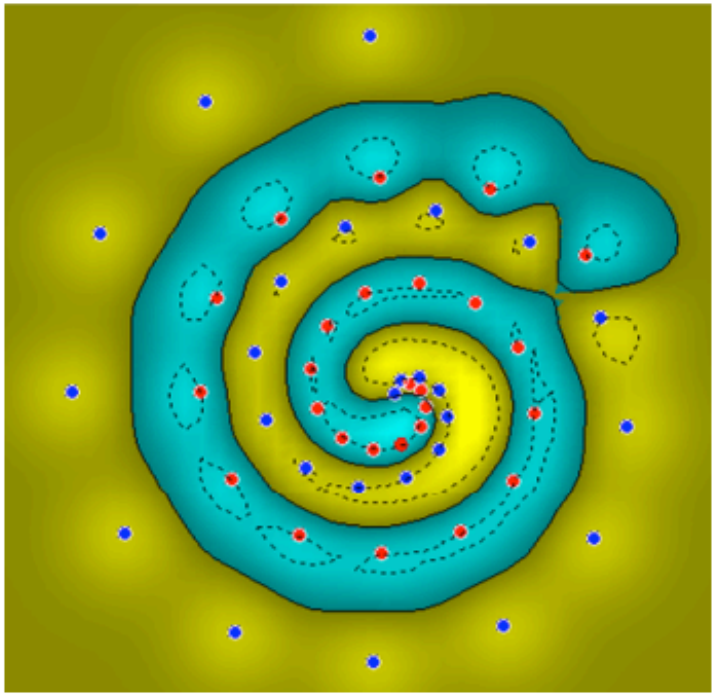
$$b = y_i - \sum_j \alpha_j y_j k(\mathbf{x}_j, \mathbf{x}_i) \quad \forall i \quad C > \alpha_i > 0$$

$$\mathbf{w}^\top \phi(\mathbf{x}) + b = \sum_i \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b$$

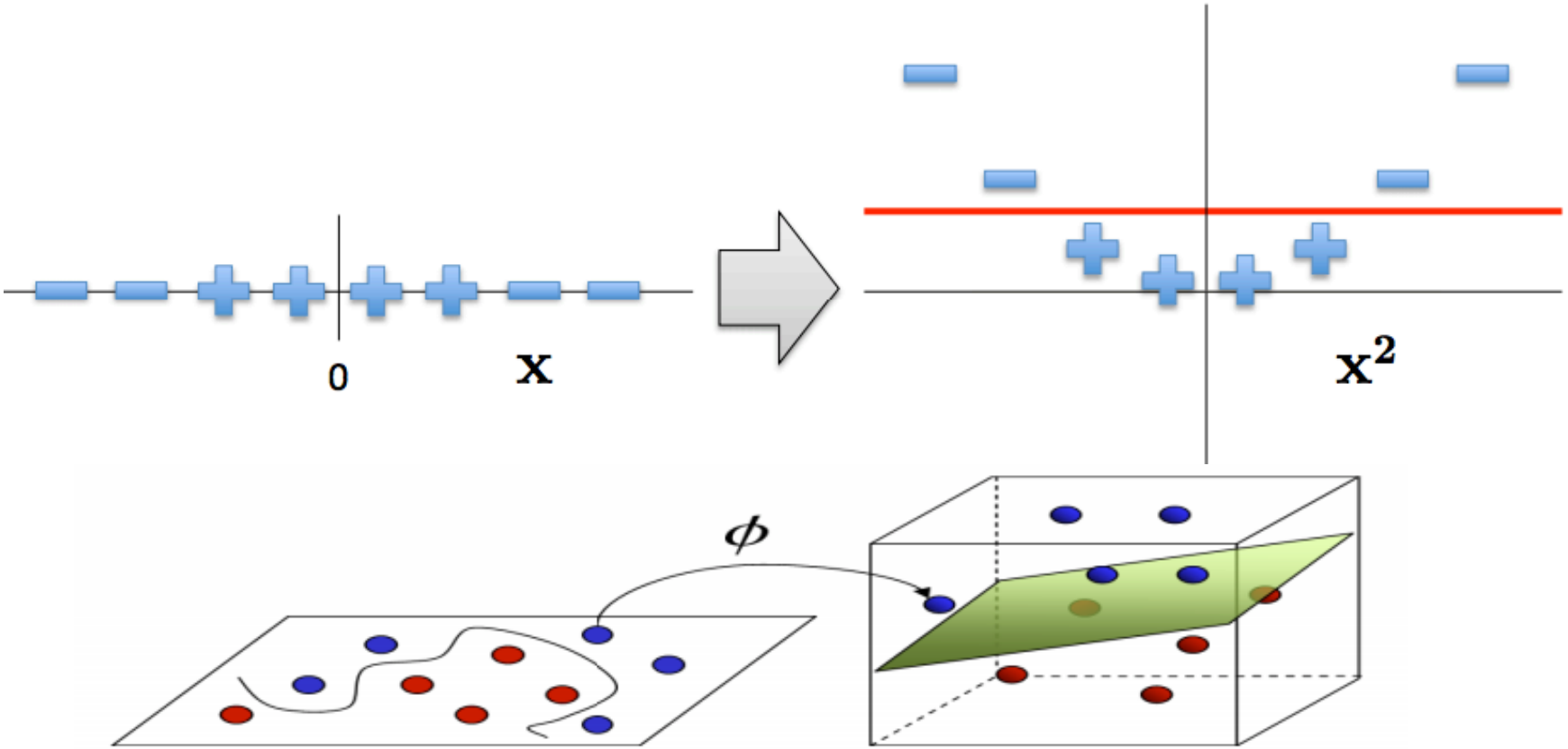
只有当 $\mathbf{x}_i$ 为支持向量的时候, $\alpha_i > 0$

为什么要使用核函数（处理线性不可分的情况）





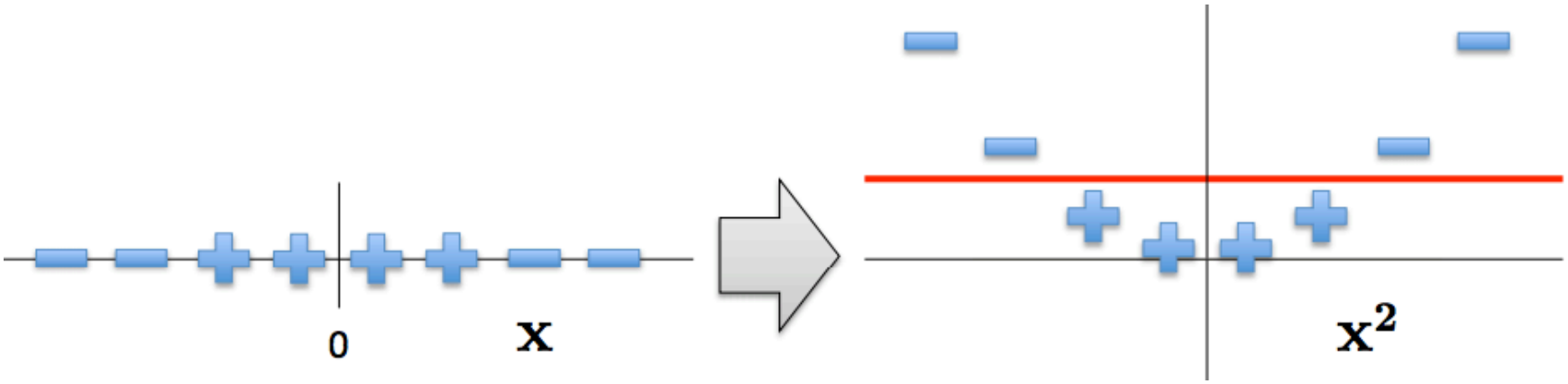
将特征映射到更高的维度



将原理的特征映射到更高的维度（续）

$$\Phi : \mathcal{X} \mapsto \hat{\mathcal{X}} = \Phi(\mathbf{x})$$

$$\Phi([x_{i1}, x_{i2}]) = [x_{i1}, x_{i2}, x_{i1}x_{i2}, x_{i1}^2, x_{i2}^2]$$



直接扩展到高纬的问题

- 一. 增大了计算量
  - 计算量与数据量和每一条数据的维度正相关
- 二. 没有办法增加到无限维

成为 Kernel 的条件: Mercer’s Theorem

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$$

Gram 矩阵:

$$G_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$$

一. 为对称矩阵

二. 为半正定矩阵 $\mathbf{z}^{\mathrm{T}}\mathrm{G}\mathbf{z} \geq 0$

$\mathbf{z} \in \mathbb{R}^n$

常用的 Kernel

多项式核（Polynomial Kernel）

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\langle \mathbf{x}_i, \mathbf{x}_j \rangle + c)^d$$

- C ≥ 0 控制低阶项的强度

- 特殊情况, 当 C = 0, d =1 成为线性核(Linear Kernel), 就于无核函数的SVM一样

$$K(\mathbf{x}_i, \mathbf{x}_j) = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$$

多项式核（Polynomial Kernel）举例

$$\mathbf{x}_i = [x_{i1}, x_{i2}] \qquad \mathbf{x}_j = [x_{j1}, x_{j2}]$$

$$\begin{aligned} K(\mathbf{x}_i, \mathbf{x}_j) &= \langle \mathbf{x}_i, \mathbf{x}_j \rangle^2 \\ &= (x_{i1}x_{j1} + x_{i2}x_{j2})^2 \\ &= (x_{i1}^2 x_{j1}^2 + x_{i2}^2 x_{j2}^2 + 2x_{i1}x_{i2}x_{j1}x_{j2}) \\ &= \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle \end{aligned}$$

$$\Phi(\mathbf{x}_i) = [x_{i1}^2, x_{i2}^2, \sqrt{2}x_{i1}x_{i2}]$$

$$\Phi(\mathbf{x}_j) = [x_{j1}^2, x_{j2}^2, \sqrt{2}x_{j1}x_{j2}]$$

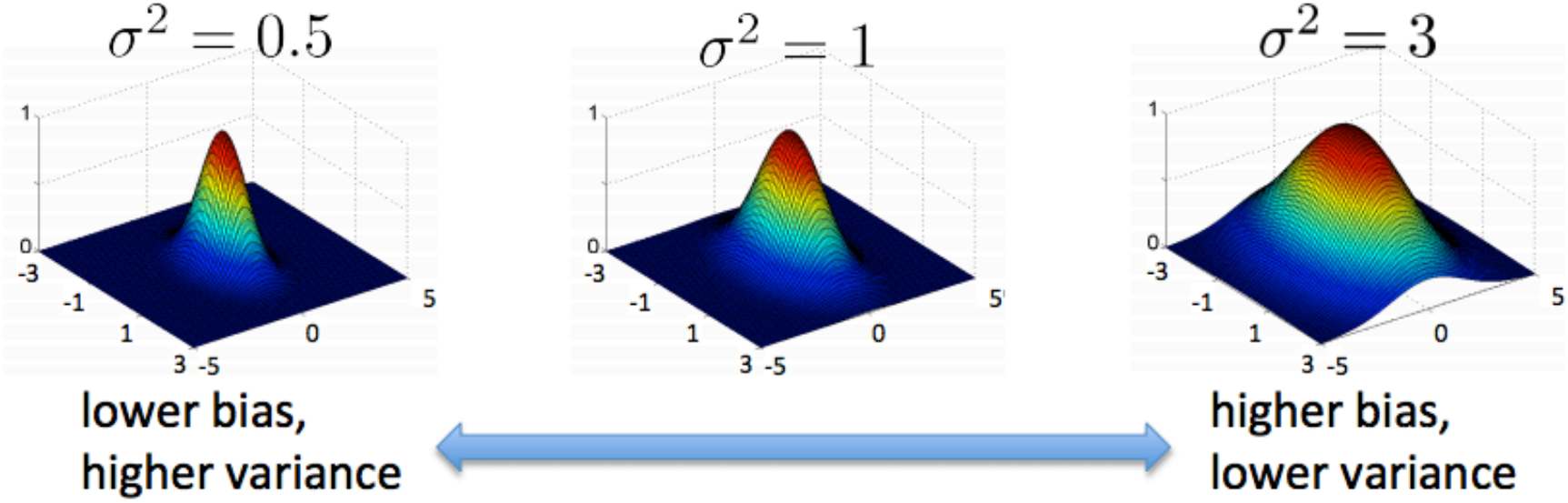
高斯核（Gaussian Kernel），也称为 Radial Basis Function（RBF）Kernel

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right)$$

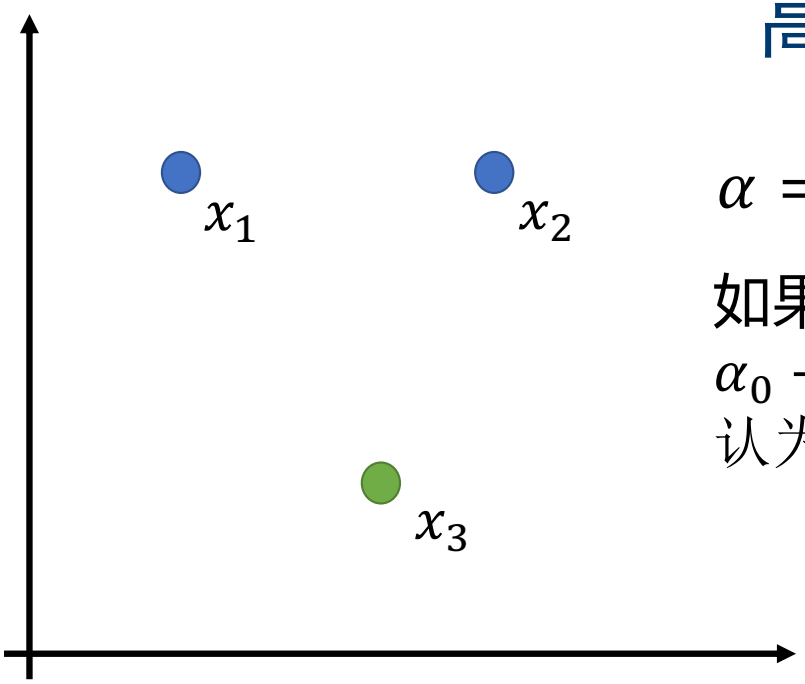
当$\mathbf{x}_i = \mathbf{x}_j$，值为1，当 x_i 与 x_j 距离增加，值倾向于0

使用高斯核之前需要将特征正规化

高斯核（Gaussian Kernel）参数的意义



高斯核举例



$$\alpha = [-0.5, 1.0, 1.0, 0.0]$$

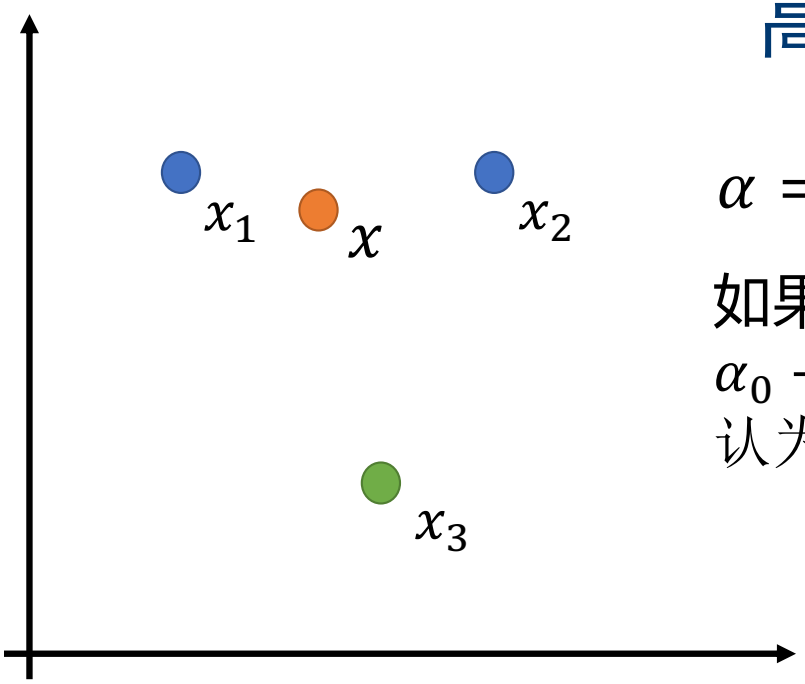
如果

$$\alpha_0 + \alpha_1 K(x, x_1) + \alpha_2 K(x, x_2) + \alpha_3 K(x, x_3) \geq 0,$$

认为输出1

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right)$$

高斯核举例



$$\alpha = [-0.5, 1.0, 1.0, 0.0]$$

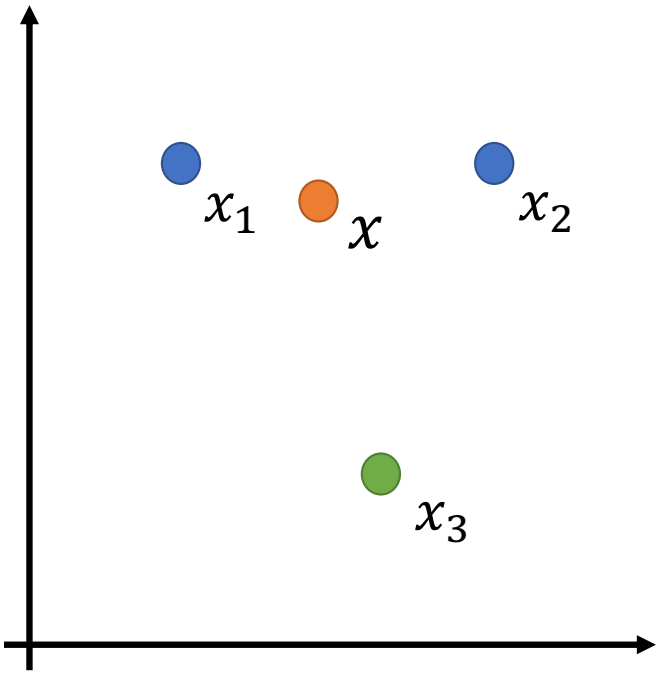
如果

$$\alpha_0 + \alpha_1 K(x, x_1) + \alpha_2 K(x, x_2) + \alpha_3 K(x, x_3) \geq 0,$$

认为输出1

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right)$$

高斯核举例



$\alpha = [-0.5, 1.0, 1.0, 0.0]$

如果

$\alpha_0 + \alpha_1 K(x, x_1) + \alpha_2 K(x, x_2) + \alpha_3 K(x, x_3) \geq 0,$

认为输出1

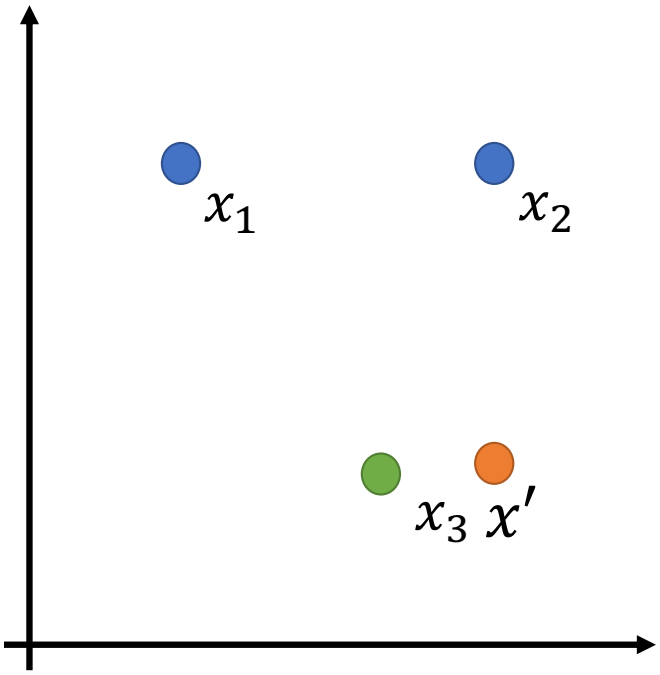
因为 x 接近 x_1, 所以 $K(x, x_1) \approx 1$, 其他情况≈ 0

$\alpha_0 + \alpha_1 K(x, x_1) + \alpha_2 K(x, x_2) + \alpha_3 K(x, x_3)$

$= -0.5 + 1 * 1 + 1 * 0 + 0 * 0 = 0.5 \geq 0$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right)$$

高斯核举例



$\alpha = [-0.5, 1.0, 1.0, 0.0]$

如果

$\alpha_0 + \alpha_1 K(x, x_1) + \alpha_2 K(x, x_2) + \alpha_3 K(x, x_3) \geq 0,$

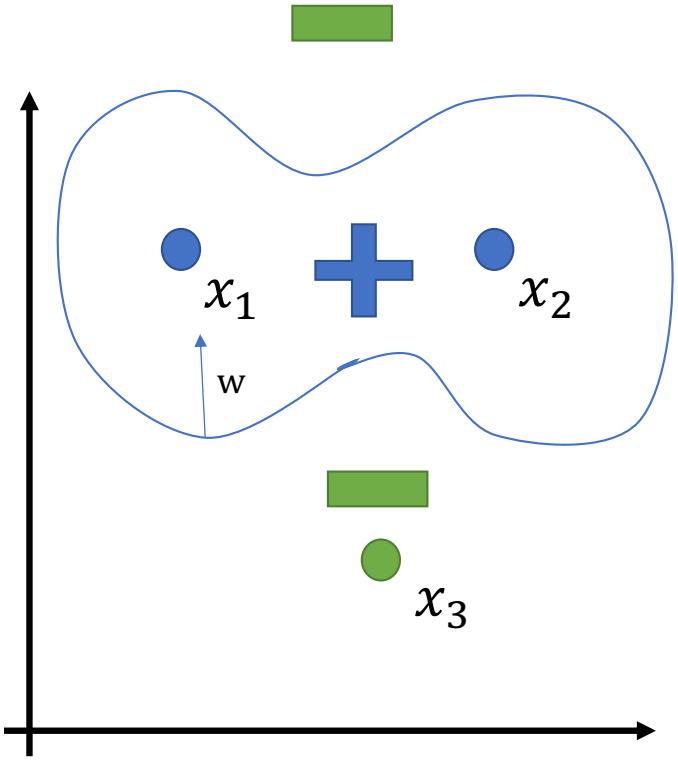
认为输出1

因为 x' 接近 x_3, 所以 $K(x', x_3) \approx 1$, 其他情况≈ 0

$\alpha_0 + \alpha_1 K(x', x_1) + \alpha_2 K(x', x_2) + \alpha_3 K(x', x_3)$

$= -0.5 + 1 * 0 + 1 * 0 + 0 * 1 = -0.5 \leq 0$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right)$$



高斯核举例

$$\alpha = [-0.5, 1.0, 1.0, 0.0]$$

如果

$$\alpha_0 + \alpha_1 K(x, x_1) + \alpha_2 K(x, x_2) + \alpha_3 K(x, x_3) \geq 0,$$

认为输出1

大致边界

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2^2}{2\sigma^2}\right)$$

Sigmoid Kernel

- 此时的SVM等价于一个没有隐含层(Hidden Layer)的简单神经网络

$$K(\mathbf{x}_i, \mathbf{x}_j) = \tanh\left(\alpha \mathbf{x}_i^{\mathsf{T}} \mathbf{x}_j + c\right)$$

Cosine Similarity Kernel

- 常用于衡量两段文字的相似性
- 相当于衡量两个向量的余弦相似度(向量夹角的余弦值)

$$K(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i^\top \mathbf{x}_j}{\|\mathbf{x}_i\| \ \|\mathbf{x}_j\|}$$

Chi-squared Kernel

- 常用于计算机视觉
- 衡量两个概率分布的相似性
- 输入数据必须是非负的, 并且使用了L1 归一化 (L1 Normalized)

$$K(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathbf{x}_i^{\mathsf{T}} \mathbf{x}_j}{\|\mathbf{x}_i\| \ \|\mathbf{x}_j\|}$$

求解 SVM

- 求解 w, b
  - 二次规划 (Quadratic Programming), 经典运筹学的最优化问题, 可以在多项式时间内求得最优解

坐标轮换法
(Coordinate Descent)

Coordinate Descent坐标轮换法简介

- 坐标轮换法是每次搜索只允许一个变量变化， 其余变量保持不变，即沿坐标方向轮流进行搜索 的寻优方法。它把多变量的优化问题轮流地转化成单变量(其余变量视为常量)的优化问题， 因此又称这种方法为变量轮换法。

- 在搜索过程中可以不需要目标函数的导数,只需目标函数值信息。这比利用目 标函数导数信息建立搜索方向的方法要简单得多。

Coordinate Descent 基本原理

- 将多维的无约束优化问题转化为一系列一维的最优化问题。每次搜索只允许一个 变量变化，其余变量保持不变，即沿着n个线性无关的方向（通常坐标方向）轮流进行搜索。

每一轮的迭代公式：

$$\boldsymbol{x}_i^{(k)} = \boldsymbol{x}_{i-1}^{(k)} + \alpha_i^{(k)} \boldsymbol{d}_i^{(k)}$$

坐标方向 $\boldsymbol{d}_i^{(k)} = \boldsymbol{e}_i$

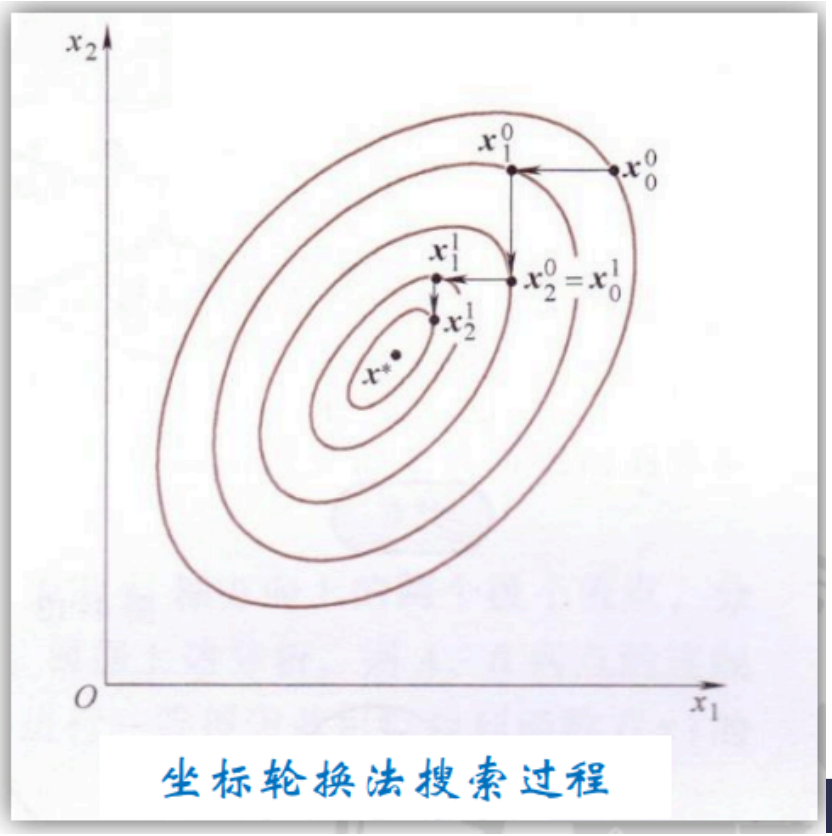
轮次 $k = 0,1,2,\cdots$

坐标 $i = 1,2,\cdots,n$

收敛判据：

$$\left\| \boldsymbol{x}_n^{(k)} - \boldsymbol{x}_0^{(k)} \right\| \le \varepsilon$$

$$\boldsymbol{x}^* \leftarrow \boldsymbol{x}_n^{(k)}$$



坐标轮换法搜索过程

基本步骤

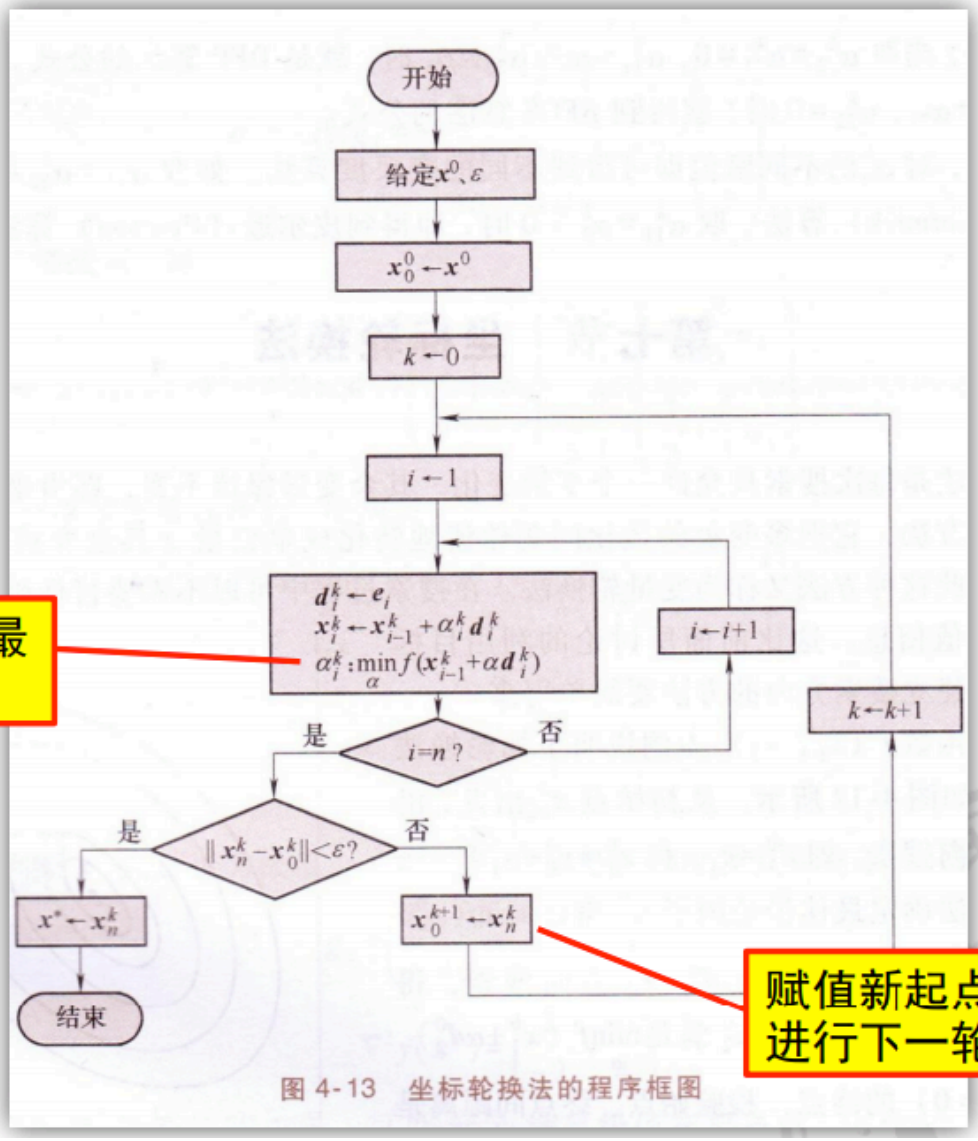


图 4-13　坐标轮换法的程序框图

用一维搜索最佳步长

赋值新起点，进行下一轮

SVM Dual

$$
\begin{aligned}
\max_{\alpha} \quad & W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle \\
\text{subject to} \quad & 0 \leq \alpha_i \leq C, \quad i = 1, \ldots, m \\
& \sum_{i=1}^{m} \alpha_i y^{(i)} = 0
\end{aligned}
$$

KKT Conditions: 满足以下条件的α_i 的集合就是优化问题的解:

$$
\begin{aligned}
\alpha_i = 0 \quad &\Rightarrow \quad y^{(i)}(w^T x^{(i)} + b) \geq 1 \\
\alpha_i = C \quad &\Rightarrow \quad y^{(i)}(w^T x^{(i)} + b) \leq 1 \\
0 < \alpha_i < C \quad &\Rightarrow \quad y^{(i)}(w^T x^{(i)} + b) = 1.
\end{aligned}
$$

SMO 算法理解（1）

- 重复直到收敛{
  - 1. 选择下一步需要优化的 α_i 和 α_j (使用启发式的方式选择α_i 和 α_j使得向最大值进发最快)
  - 保持其它的 α 值不变, 仅仅通过改变 α_i 和 α_j 来优化 W(α)
- }

- 使用 KKT条件来判断是否收敛.

SMO 算法理解（2）

- 假设我们需要优化α_1 和 α_2, 根据约束条件 $\sum_{i=1}^{n} \alpha_i y^{(i)} = 0$, 可以推导

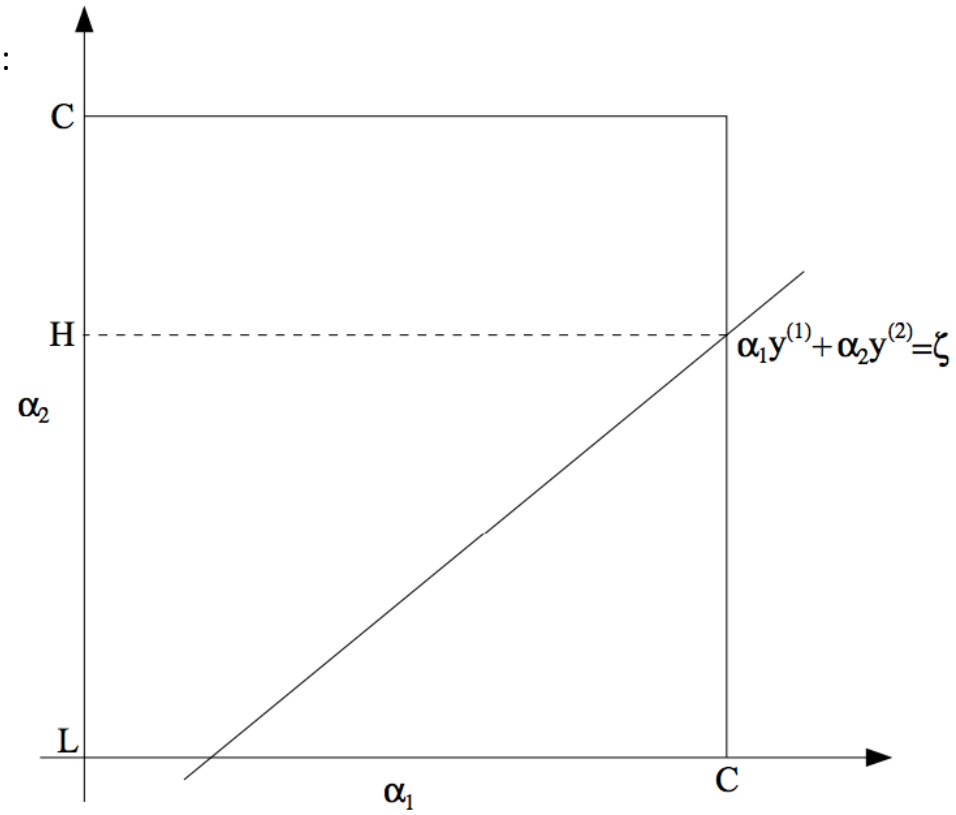
$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = -\sum_{i=3}^{n} \alpha_i y^{(i)}$$

- 由于等式右边保持固定, 可以使用常数代替, 可以转化为

$$\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = \zeta$$

SMO 算法理解（3）

α_1 和 α_2的约束关系可以显示如下:



SMO 算法理解（4）

- 根据 $\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = \zeta$，可以把 α_1 写为关于 α_2 的函数:

$$\alpha_1 = (\zeta - \alpha_2 y^{(2)}) y^{(1)}$$

- 原来的优化函数变为:

$$W(\alpha_1, \alpha_2, \ldots, \alpha_n) = W((\zeta - \alpha_2 y^{(2)}) y^{(1)}, \alpha_2, \ldots, \alpha_n)$$

- $\alpha_3, \ldots, \alpha_n$ 可以看为常数，于是优化函数可以写为 $\mathrm{a}\alpha_2^2 + \mathrm{b}\alpha_2 + \mathrm{c}$ 的形式，如果我们忽略α_1 和 α_2的约束关系，可以对二次函数求导，得到最优解. 假设 $\alpha_2^{new,unclipped}$为 α_2的解，那么α_2的新的值应该为. 同理可以求α_2的新的值.

$$\alpha_2^{new} = \begin{cases} H & \text{if } \alpha_2^{new,unclipped} > H \\ \alpha_2^{new,unclipped} & \text{if } L \le \alpha_2^{new,unclipped} \le H \\ L & \text{if } \alpha_2^{new,unclipped} < L \end{cases}$$

SVM 对偶问题的SMO解法伪代码

- http://cs229.stanford.edu/materials/smo.pdf

SMO 算法

- 随机选择 α_i 和 α_j, 计算 $k = K(i,i) + K(j,j) - 2K(i,j)$, 如果 $k == 0$, 重新选择 α_i 和 α_j 直到 $k \neq 0$,
- $\alpha_j^{new,unc} \leftarrow \alpha_j^{old} + \frac{y_j\{E_j - E_i\}}{k}$
  - $E_m \leftarrow (\sum_{l=1}^{n} \alpha_l y_l K(l,m) + b) - y_m, m = i, j$
- 将 α_j 剪切到 $U \leq \alpha_j \leq V$ 之间
- $$\alpha_j^{new} = \begin{cases} V & if \quad \alpha_j^{new,unc} > V \\ \alpha_j^{new,unc} & if \quad U \leq \alpha_j^{new,unc} \leq V \\ U & if \quad \alpha_j^{new,unc} < U \end{cases}$$
- 其中 U, V 定义如下:
  - 如果 $y_i \neq y_j$, $\begin{cases} \mathrm{U} = \max\{0, \quad \alpha_j^{old} - \alpha_i^{old}\} \\ V = \min\{C, \quad C - \alpha_i^{old} + \alpha_j^{old}\} \end{cases}$, 如果 $y_i = y_j$, $\begin{cases} \mathrm{U} = \max\{0, \quad \alpha_i^{old} + \alpha_j^{old} - C\} \\ V = \min\{C, \quad \alpha_i^{old} + \alpha_j^{old}\} \end{cases}$
- $\alpha_i^{new} = \alpha_i^{old} + y_i y_j (\alpha_j^{old} - \alpha_j^{new})$
- $$b = \begin{cases} b_1 = b - E_i - y^i(\alpha_i^{new} - \alpha_i^{old})K(i,i) - y^j(\alpha_j^{new} - \alpha_j^{old})K(i,j) & if \ 0 < \alpha_i^{new} < C \\ b_2 = b - E_j - y^i(\alpha_i^{new} - \alpha_i^{old})K(i,j) - y^j(\alpha_j^{new} - \alpha_j^{old})K(j,j) & if \ 0 < \alpha_j^{new} < C \\ (b_1 + b_2)/2 \quad 其它 \end{cases}$$

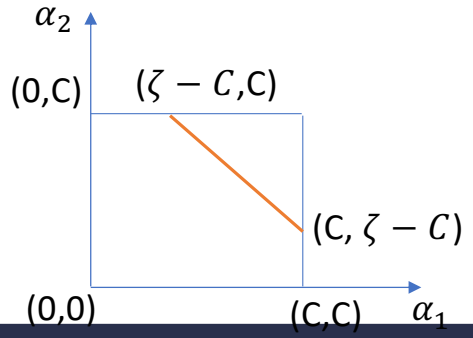
证明 SMO 中α^{new}的上下界

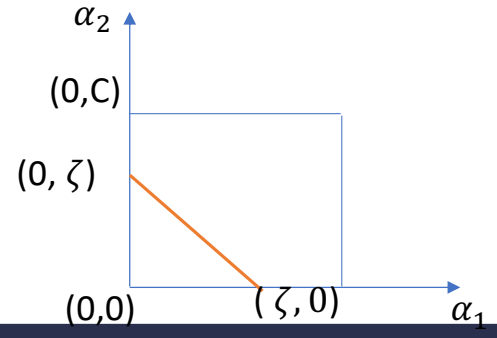
- 不失一般性, 我们假设选中α_1和α_2进行优化, 根据SVM Dual形式的定义(如下), 知道α_2^{new}的值一定在[0, C]之间.

$$\begin{aligned} \max_\alpha \quad & W(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle \\ \text{s.t.} \quad & 0 \le \alpha_i \le C, \quad i = 1, \ldots, n \\ & \sum_{i=1}^{n} \alpha_i y^{(i)} = 0, \end{aligned}$$

根据约束条件 $\sum_{i=1}^{n} \alpha_i y^{(i)} = 0$, 可以得到 $\alpha_1 y^{(1)} + \alpha_2 y^{(2)} = -\sum_{i=3}^{n} \alpha_i y^{(i)}$ 当我们优化α_1和α_2时, 假设$\alpha_3, \ldots, \alpha_n$都为常数, 于是, 可以使用常数$\zeta$代表 $-\sum_{i=3}^{n} \alpha_i y^{(i)}$, 于是α_1和α_2的关系可以表达为$\alpha_1 y^1 + \alpha_2 y^2 = \zeta$. 也即是$\alpha_2$的值在一个正方形区域内的直线上. 我们分两种情形讨论:

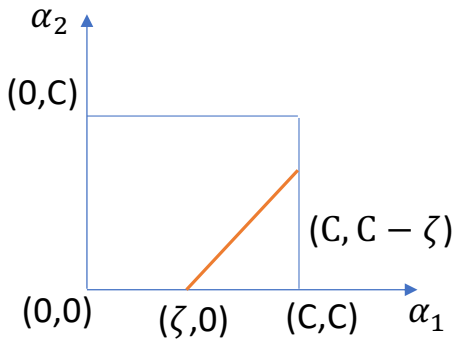
- 1. 当$y^1 == y^2$时, $\alpha_1 y^1 + \alpha_2 y^2 = \alpha_1 + \alpha_2 = \zeta$, 又分为两种子情况讨论:
  - 1.1 当$\zeta > C$, 如下左图所示, $\max \alpha_2 = C$, $\min \alpha_2 = \zeta - C = \alpha_1 + \alpha_2 - C$
  - 1.2当$\zeta < C$, 如下右图所示, $\max \alpha_2 = \zeta = \alpha_1 + \alpha_2$, $\min \alpha_2 = 0$

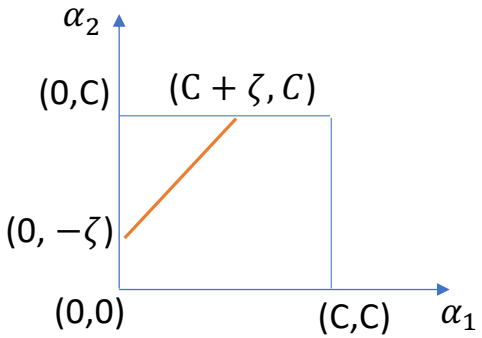




证明 SMO 中α^{new}的上下界（续）

- 2. 当$y^1 \neq y^2$时, $\alpha_1 y^1 + \alpha_2 y^2 = \alpha_1 - \alpha_2 = \zeta$, 又分为两种子情况讨论:
  - 2.1 当$\zeta > 0$, 如下左图所示, $\max \alpha_2 = C - \zeta = C - \alpha_1 + \alpha_2$, $\min\ \alpha_2 = 0$
  - 2.2当$\zeta < 0$, 如下右图所示, $\max \alpha_2 = C$, $\min \alpha_2 = -\zeta = -\alpha_1 + \alpha_2$





$$\text{如果} y^1 \neq y^2, \begin{cases} \text{下届} = \max\{0, \quad \alpha_j^{old} - \alpha_i^{old}\} \\ \text{上届} = \min\{C, \quad C - \alpha_i^{old} + \alpha_j^{old}\} \end{cases}$$

$$\text{如果} y^1 == y^2, \begin{cases} \text{下届} = \max\{0, \quad \alpha_i^{old} + \alpha_j^{old} - C\} \\ \text{上届} = \min\{C, \quad \alpha_i^{old} + \alpha_j^{old}\} \end{cases}$$

SVM SMO代码

- https://github.com/yuan776/SVM-w-SMO/blob/master/SVM.py

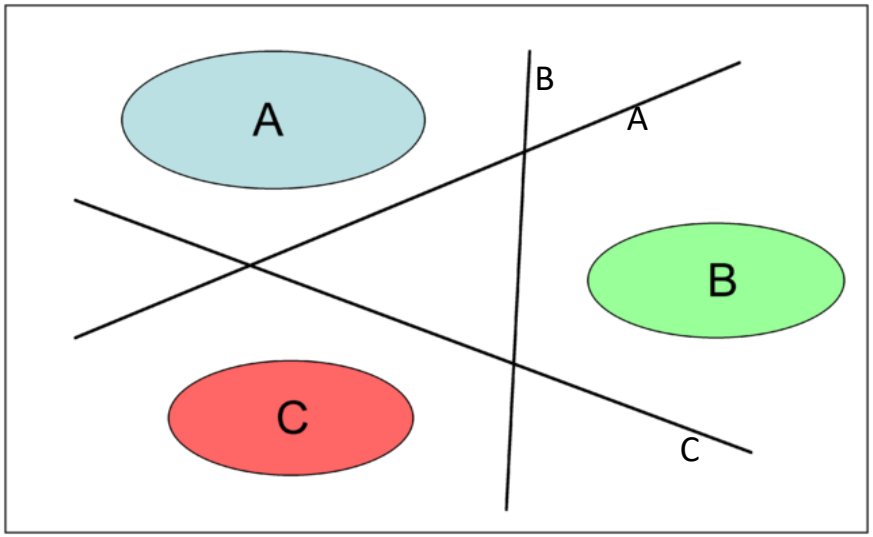
扩展SVM到支持多个类别

- 两种方法:
- 1. OVR (one versus rest): 对于K个类别的情况, 训练K个SVM, 第 *j* 个SVM用于判断任意条数据是属于类别 *j* 还是属于类别非 *j*. 预测的时候, 具有最大值的 $\mathrm{w}_i^T \boldsymbol{x} + b_i$ 表示给定的数据 $\boldsymbol{x}$ 属于类别 *i*.

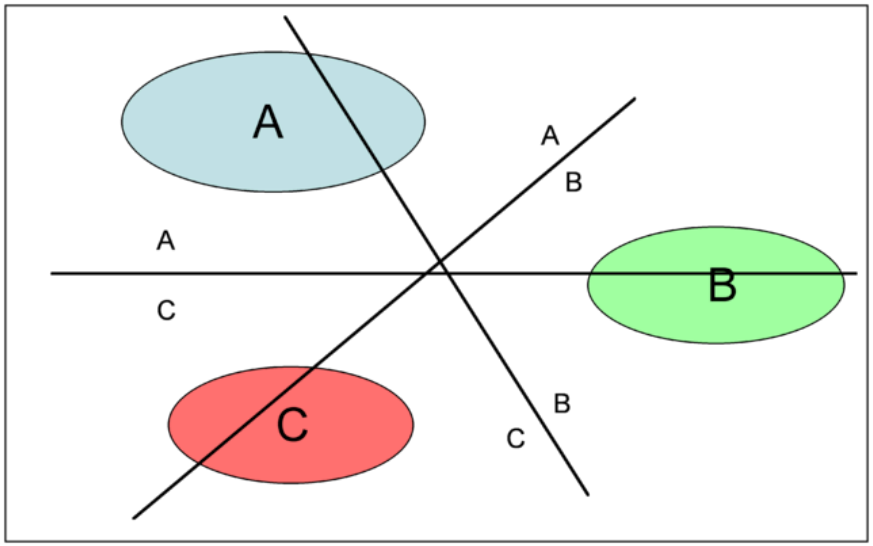
- 2. OVO (one versus one), 对于K个类别的情况, 训练K * (K-1) /2 个SVM, 每一个SVM只用于判读任意条数据是属于K中的特定两个类别. 预测的时候, 使用K * (K-1) /2 个SVM做 K * (K-1) /2 次预测, 使用计票的方式决定数据被分类为哪个类别的次数最多, 就认为数据 $\boldsymbol{x}$ 属于此类别.

扩展SVM到支持多个类别

One versus All



One versus One



实例演示

总结

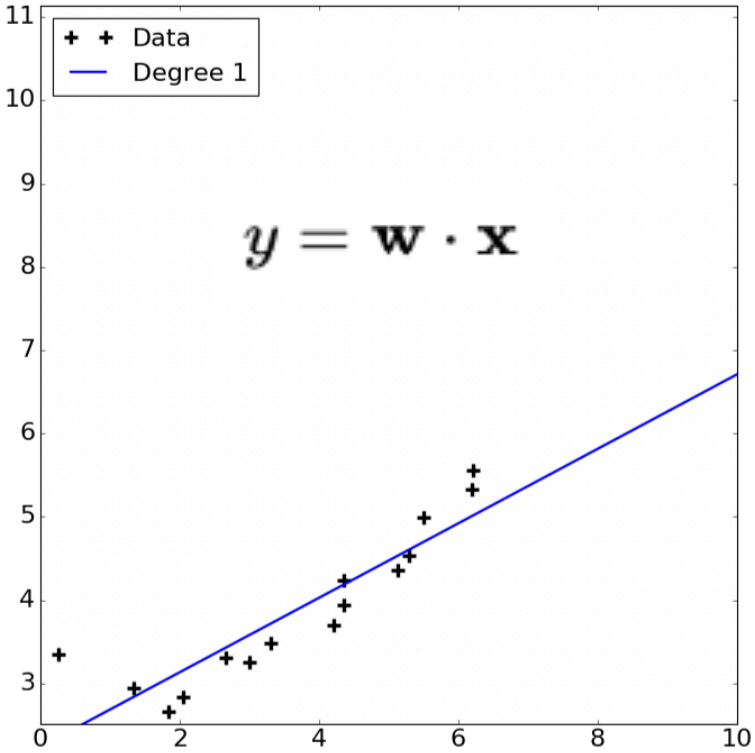
- SVM 专注于找最优分界线, 用于减小过拟合
- Kernel Trick的应用使得 SVM 可以高效的用于非线性可分的情况

- 优势
  - 理论非常完美
  - 支持不同Kernel, 用于调参

- 缺点
  - 当数据量特别大时, 训练比较慢

Kernel Linear Regression

回顾Linear Regression



Linear Regression: Dual Form

- Primal form:
  - 模型: $\widehat{f(x)} = \sum_{i=1}^{M} w_i x_i = \mathrm{w}^{\mathrm{T}} x = \langle w, x \rangle$
  - 优化问题: $w = arg\min_{w} \|y - Xw\|^2 + \lambda \|w\|^2$
  - 解: $w = (X^T X + \lambda I)^{-1} X^T y$
- Dual form:
  - 已知 $w = (X^T X + \lambda I)^{-1} X^T y = X^T (XX^T + \lambda I)^{-1} y$
  - 令 $\alpha = (XX^T + \lambda I)^{-1} y$
  - $\mathrm{w} = \sum_{l=1}^{N} \alpha_l x^l = X^T \alpha$
  - 模型: $\widehat{f(x)} = \mathrm{w}^{\mathrm{T}} x = \sum_{l=1}^{N} \alpha_l \langle x, x^l \rangle$
  - 解: $\alpha = (XX^T + \lambda I)^{-1} y$
  - Kernel Trick:
    - $\alpha = (K + \lambda I)^{-1} y$
    - $\widehat{f(x)} = \sum_{l=1}^{N} \alpha_l k(x, x^l)$

Note that

$$(X^T X + \lambda I) X^T = X^T X X^T + \lambda X^T = X^T (XX^T + \lambda I)$$

Multiplying each part of the equation by $(X^T X + \lambda I)^{-1}$ at the left and $(XX^T + \lambda I)^{-1} y$ at the right, we have

$$X^T (XX^T + \lambda I)^{-1} y = (X^T X + \lambda I)^{-1} X^T y$$

Kernel Linear Regression(Radial Basis Function kernel)

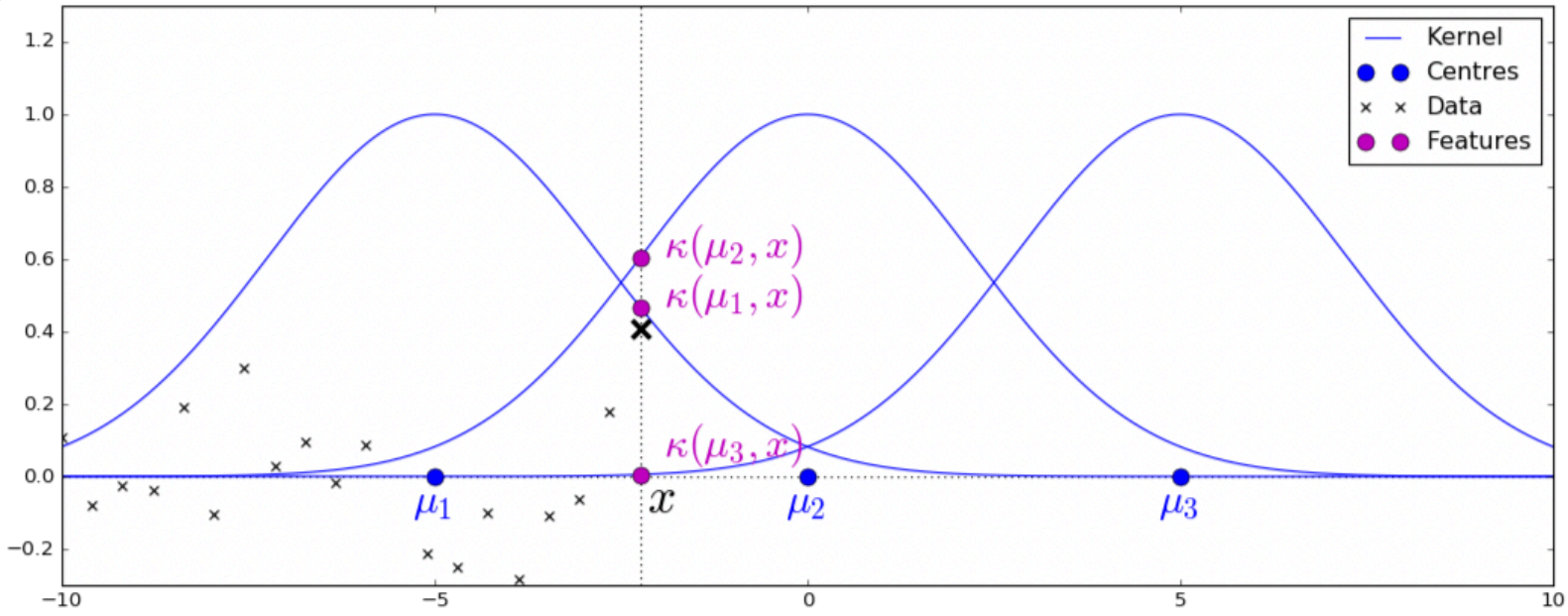
$$y = w_0 + w_1\kappa(\mu_1, \mathbf{x}) + \cdots + w_M\kappa(\mu_M, \mathbf{x}) + \epsilon = \mathbf{w} \cdot \phi(\mathbf{x}) + \epsilon$$

$$\phi(\mathbf{x}) = [1, \kappa(\mu_1, \mathbf{x}), \ldots, \kappa(\mu_M, \mathbf{x})]$$

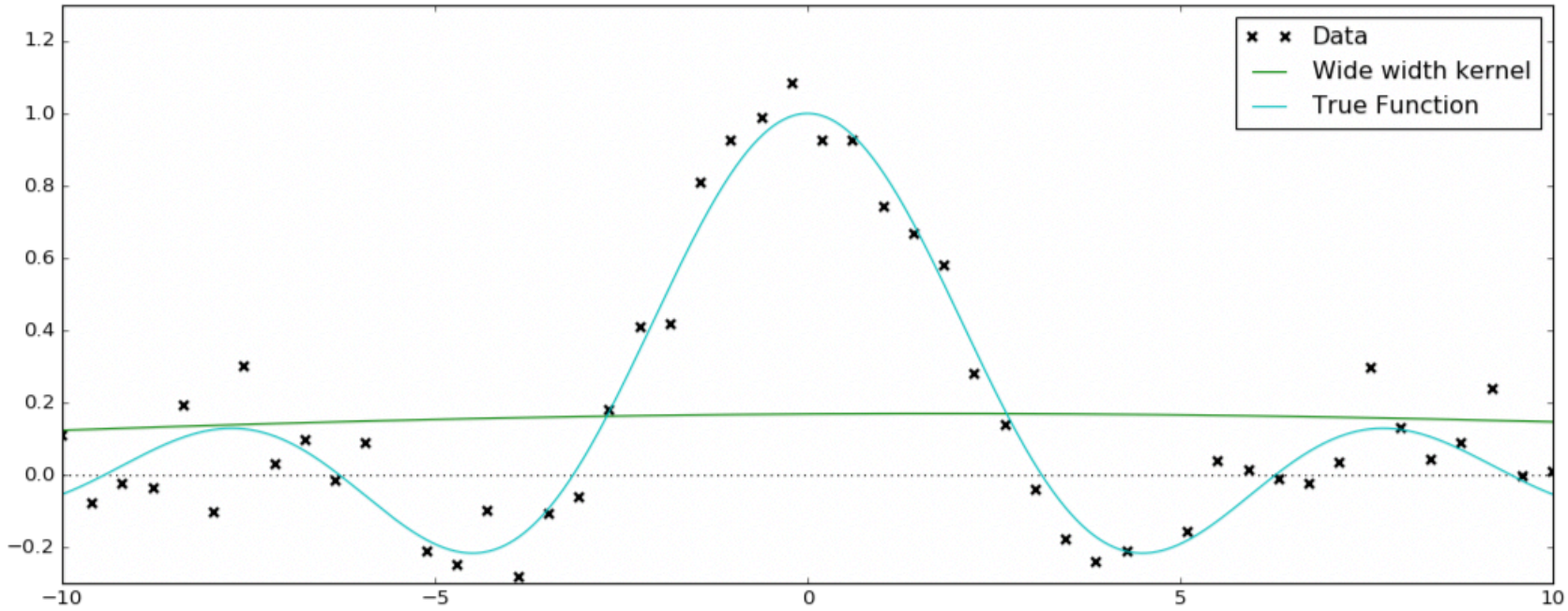
$$\kappa(\mathbf{x}', \mathbf{x}) = \exp\left(-\gamma\|\mathbf{x} - \mathbf{x}'\|^2\right)$$

选择Kernel的中心为 $\mu_1, \mu_2, \cdots, \mu_M$，也可以选择 X 本身中的一些点

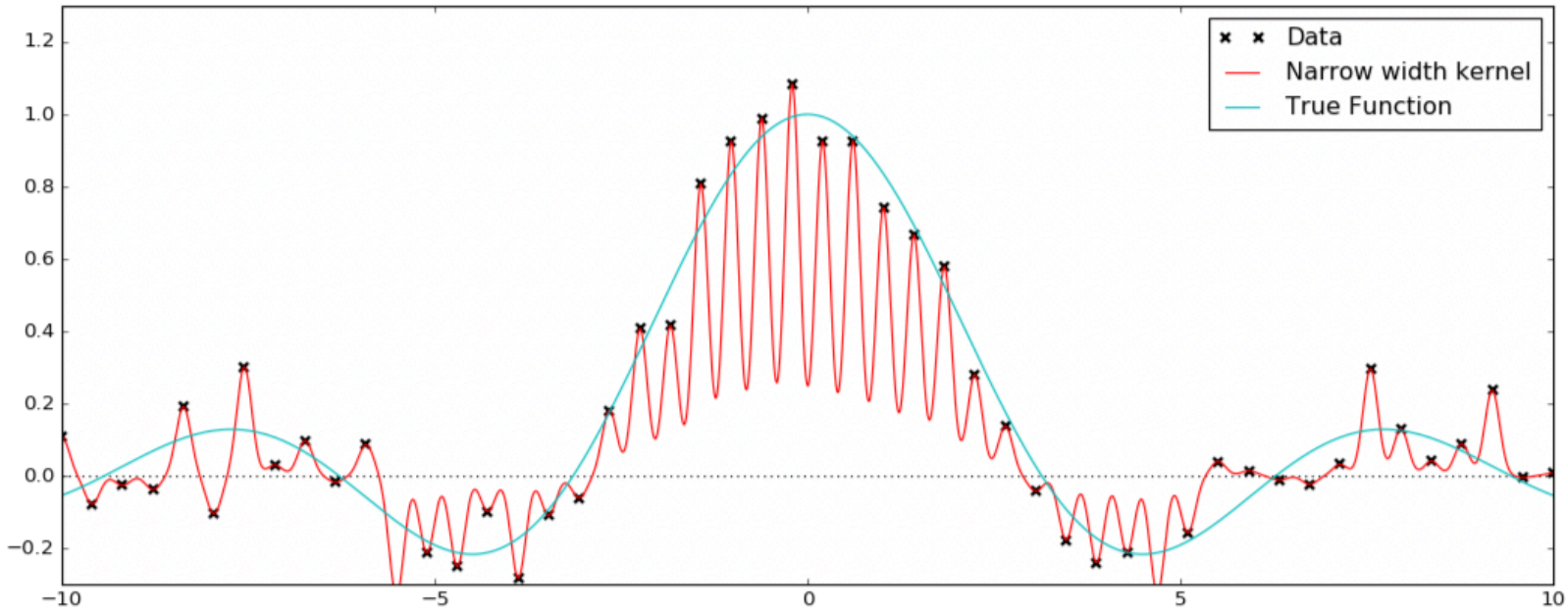
当γ太大, 容易overfitting
当γ太小, 容易underfitting



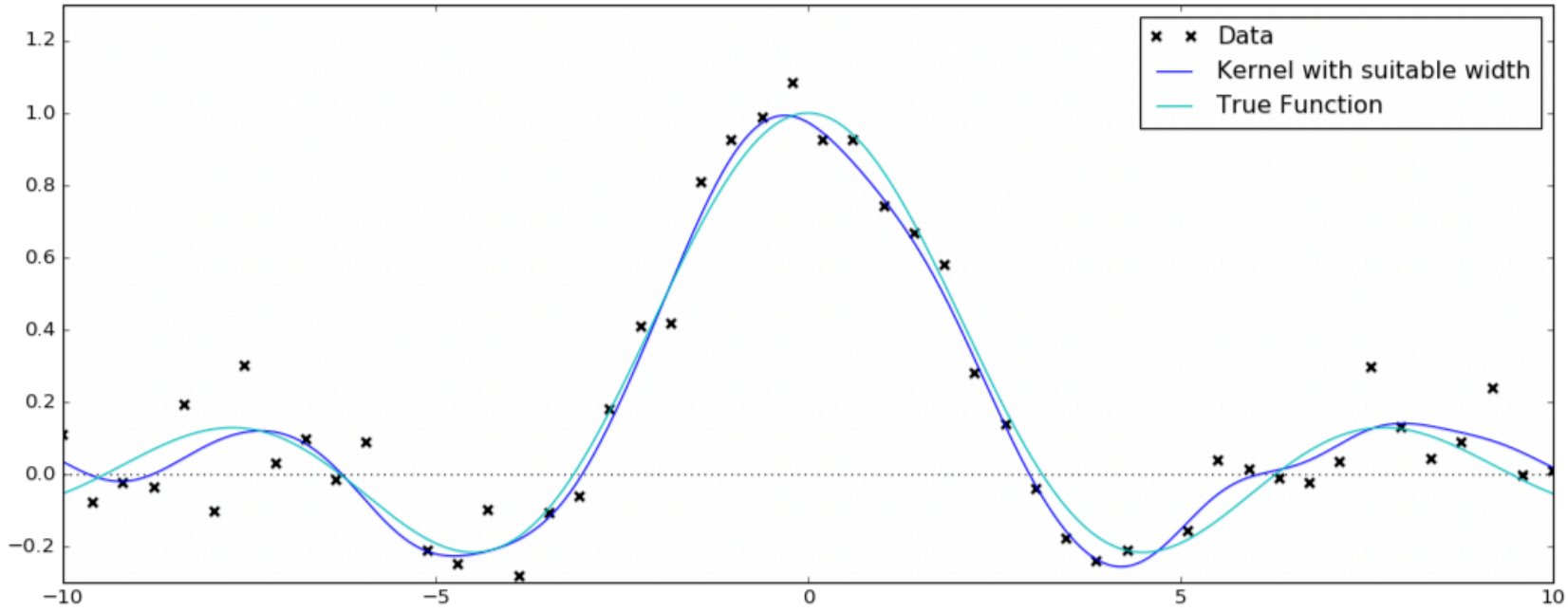
当γ太小，对应RBF的width 参数大，容易underfitting



当γ太大，对应RBF的width 参数小，容易overfitting



当γ适中，对应RBF的width参数适中，good fitting



Kernel PCA

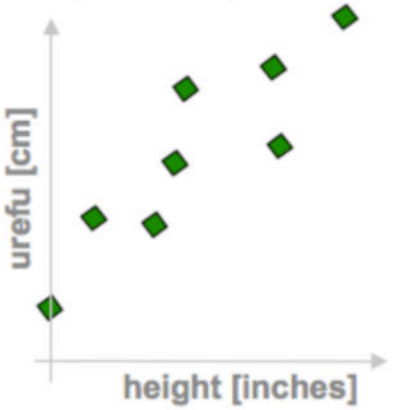
回顾 PCA

- 对于中心化后的数据进行分析，即 $\frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i = \mathbf{0}$

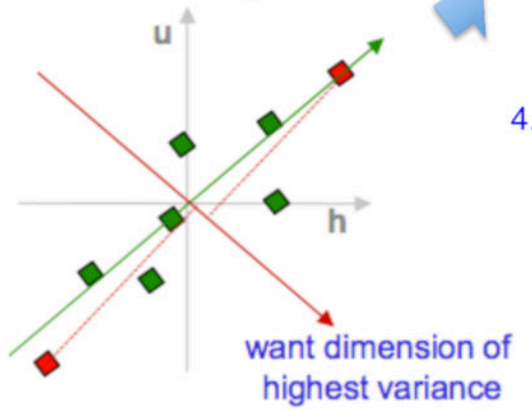
对其协方差矩阵 $C = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i\mathbf{x}_i^\top$ 进行特征分解 $\lambda\mathbf{v} = C\mathbf{v}$

PCA in a nutshell

1. correlated hi-d data
("urefu" means "height" in Swahili)



2. center the points



3. compute covariance matrix

$$\begin{array}{c} \\ h \\ u \end{array}\begin{array}{c} \begin{array}{cc} h & u \end{array} \\ \begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \end{array} \rightarrow \mathrm{cov}(h,u) = \frac{1}{n}\sum_{i=1}^{n} h_i u_i$$

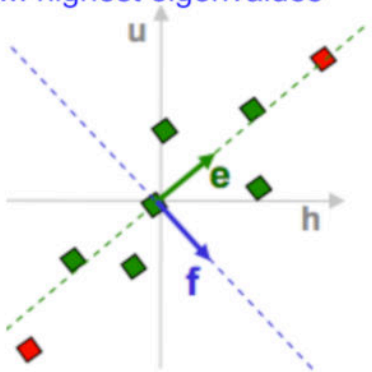
4. eigenvectors + eigenvalues

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \begin{pmatrix} e_h \\ e_u \end{pmatrix} = \lambda_e \begin{pmatrix} e_h \\ e_u \end{pmatrix}$$

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \begin{pmatrix} f_h \\ f_u \end{pmatrix} = \lambda_f \begin{pmatrix} f_h \\ f_u \end{pmatrix}$$

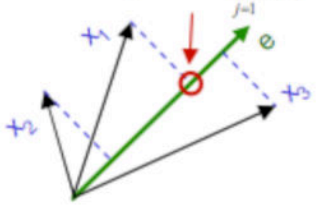
```
eig(cov(data))
```

5. pick m<d eigenvectors
w. highest eigenvalues

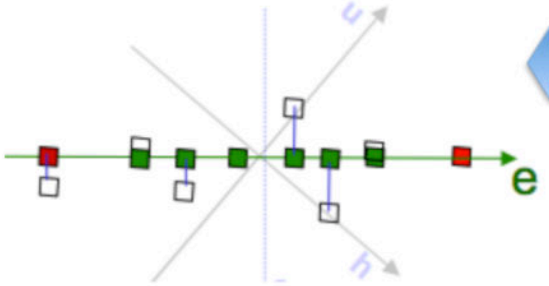


6. project data points to
those eigenvectors

$$x_e' = x^T e = \sum_{j=1}^{d} x_j e_j$$



7. uncorrelated low-d data



Kernel PCA (1)

- 一般而言，*m* 个数据点在 *d* < *m* 维空间中是线性不可分的，但它们在 *d* >= *m* 维空间中则是几乎必然线性可分的。这也意味着，如果我们能将 *m* 个数据点映射到一个 >= *m* 维空间中，就能很容易地构建一个超平面将数据点作任意分类.

Kernel PCA (2)

- 假设数据在特征空间的均值为零, 即 $\sum_{i=1}^{m} \emptyset(x_i) = 0, \emptyset(x_i) \in R^N, x_i \in R^D$
- 协方差矩阵为

$$C = \frac{1}{m} \sum_{i=1}^{m} \emptyset(x_i) \emptyset(x_i)^T$$

- 特征向量为:

$$Cv_j = \lambda_j v_j, j = 1, \ldots, N$$

- 为了避免映射到特征空间, 使用Kernels:

$$K(x_i, x_k) = \emptyset(x_i)^T \emptyset(x_k)$$

Kernel PCA (3)

- PCA 等式使用Kernels

$$\frac{1}{m}\sum_{i=1}^{m}\emptyset(x_i)\emptyset(x_i)^T v_j = \lambda_j v_j, j = 1, \dots, N \quad (1)$$

- 特征向量可以表达为特征空间的值的线性组合:

$$v_j = \sum_{l=1}^{m}\alpha_{jl}\emptyset(x_l)$$

- 寻找特征向量等价于寻找系数 $\alpha_{ji}, \mathrm{j} = 1, \dots, N, i = 1, \dots . m$
- 带入(1), 得到:

$$\frac{1}{m}\sum_{i=1}^{m}\emptyset(x_i)\emptyset(x_i)^T\left(\sum_{l=1}^{m}\alpha_{jl}\emptyset(x_l)\right) = \lambda_j\sum_{l=1}^{m}\alpha_{jl}\emptyset(x_l)$$

- 可以重写为:

$$\frac{1}{m}\sum_{i=1}^{m}\emptyset(x_i)\left(\sum_{l=1}^{m}\alpha_{jl}K(x_i, x_l)\right) = \lambda_j\sum_{l=1}^{m}\alpha_{jl}\emptyset(x_l)$$

- 上式左右都乘以$\emptyset(x_k)^T$, 可得:

$$\frac{1}{m}\sum_{i=1}^{m}\emptyset(x_k)^T\emptyset(x_i)\left(\sum_{l=1}^{m}\alpha_{jl}K(x_i, x_l)\right) = \lambda_j\sum_{l=1}^{m}\alpha_{jl}\emptyset(x_k)^T\emptyset(x_l)$$

Kernel PCA (4)

$$\frac{1}{m}\sum_{i=1}^{m}\emptyset(x_k)^T\emptyset(x_i)\left(\sum_{l=1}^{m}\alpha_{jl}K(x_i,x_l)\right)=\lambda_j\sum_{l=1}^{m}\alpha_{jl}\emptyset(x_k)^T\emptyset(x_l)$$

- 代入kernel:

$$\frac{1}{m}\sum_{i=1}^{m}K(x_k,x_i)\left(\sum_{l=1}^{m}\alpha_{jl}K(x_i,x_l)\right)=\lambda_j\sum_{l=1}^{m}\alpha_{jl}K(x_k,x_l)$$

- 上式可以简写为

$$K^2\,\alpha_j=m\lambda_jK\alpha_j$$

- 上式进一步精简为

$$K\alpha_j=m\lambda_j\alpha_j$$

- PCA的特征向量是Normalized, 即

$$v_j^T\,v_j=1\Rightarrow\sum_{k=1}^{m}\sum_{l=1}^{m}\alpha_{jl}\alpha_{jk}\emptyset(x_l)^T\emptyset(x_k)=1\Rightarrow\alpha_j^TK\alpha_j=1$$

- 上式代入$K\alpha_j=m\lambda_j\alpha_j$, 可得

$$m\lambda_j\alpha_j^T\alpha_j=\alpha_j^TK\alpha_j=1$$

- 对于一个新的数据 x, 其在principal components $v_j=\sum_{l=1}^{m}\alpha_{jl}\emptyset(x_l)$ 的投影为:

$$\emptyset(x)^Tv_j=\sum_{l=1}^{m}\alpha_{jl}\emptyset(x)^T\emptyset(x_l)=\sum_{l=1}^{m}\alpha_{jl}K(x,x_i)$$

Kernel PCA (5)

- 一般而言, $\emptyset(x_i)$的均值不会为0
- 其均值化的公式:

$$\emptyset(x_i) = \emptyset(x_i) - \frac{1}{m}\sum_{k=1}^{m}\emptyset(x_k)$$

- $\emptyset(x_i)$ 对应的kernel matrix K 为:

$$K_{ij}^{C} = K^{C}(x_i, x_j) = \left(\emptyset(x_i) - \frac{1}{m}\sum_{k=1}^{m}\emptyset(x_k)\right)\left(\emptyset(x_i) - \frac{1}{m}\sum_{k=1}^{m}\emptyset(x_k)\right)^{T}$$

$$= \emptyset_i\emptyset_j^T - \left[\frac{1}{m}\sum_{k=1}^{m}\emptyset_k\right]\emptyset_j^T - \emptyset_i\left[\frac{1}{m}\sum_{l=1}^{m}\emptyset_l\right] + \left[\frac{1}{m}\sum_{k=1}^{m}\emptyset_k\right]\left[\frac{1}{m}\sum_{l=1}^{m}\emptyset_l\right]$$

$$= K_{ij} - k(x_i)1_j^T - 1_i k(x_j)^T + k1_i1_j^T$$

其中$k(x_i) = \frac{1}{m}\sum_{l=1}^{m}K_{il}$ ， $k = \frac{1}{m^2}\sum_{i=1,j=1}^{m,m}K_{ij}$

使用Kernel PCA降维向量 t_i时, 需要计算相应的 $K_{ij}^{C'} = K^{C'}(t_i, x_j) = \left(\emptyset(t_i) - \frac{1}{m}\sum_{k=1}^{m}\emptyset(x_k)\right)\left(\emptyset(x_i) - \frac{1}{m}\sum_{k=1}^{m}\emptyset(x_k)\right)^{T}$, 可以得到

$$K^{C}(t_i, x_j) = K(t_i, x_j) - k(t_i)1_j^T - 1_i k(x_j)^T + k1_i1_j^T$$

kernel PCA 步骤总结

- 1. 选择一个kernel K
- 2. 构造正规化(normalized)的kernel matrix $\widetilde{\mathrm{K}}$, 维度为 m × m

$$\widetilde{K} = K_{ij} - k(x_i)1_j^T - 1_i k\left(x_j\right)^T + k1_i 1_j^T$$

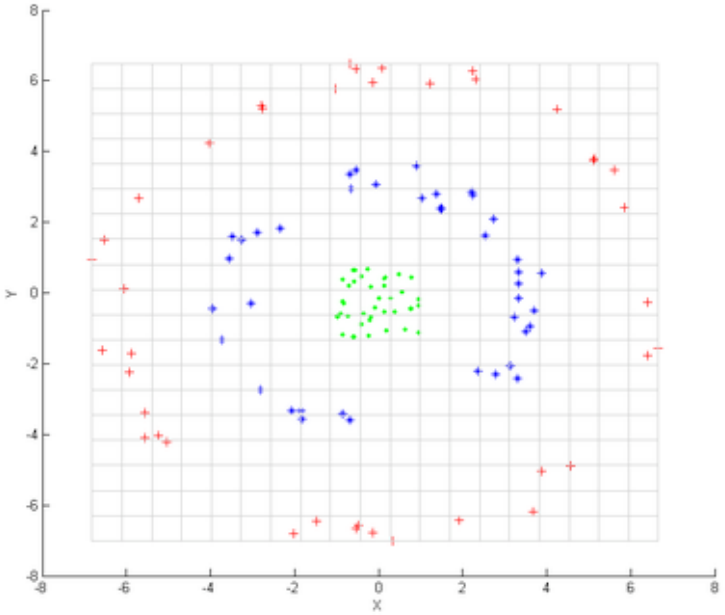
- 3. 找出 $\widetilde{\mathrm{K}}$ 的特征值 λ_j 和特征向量 $\boldsymbol{\alpha}_j$
- 4. 对于任意数据点t_i, 可以用如下方式降维:

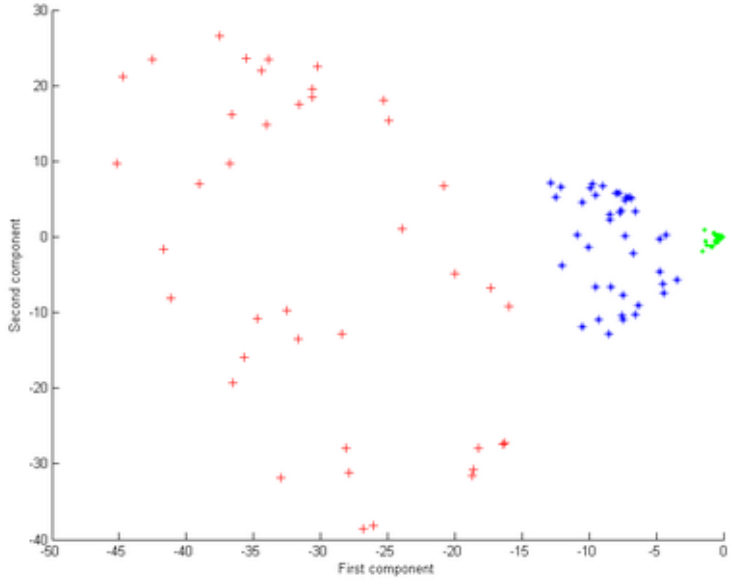
$$y_j = \sum_{i=1}^{m} \alpha_{ji} K(X, X_i), \qquad j = 1, \ldots, m$$

如果我们限制 j < m (选择前 j 个最大特征值对应的 α), 就相当于降维了

Kernel PCA 实例

$$k(\boldsymbol{x}, \boldsymbol{y}) = (\boldsymbol{x}^{\mathrm{T}}\boldsymbol{y} + 1)^2$$

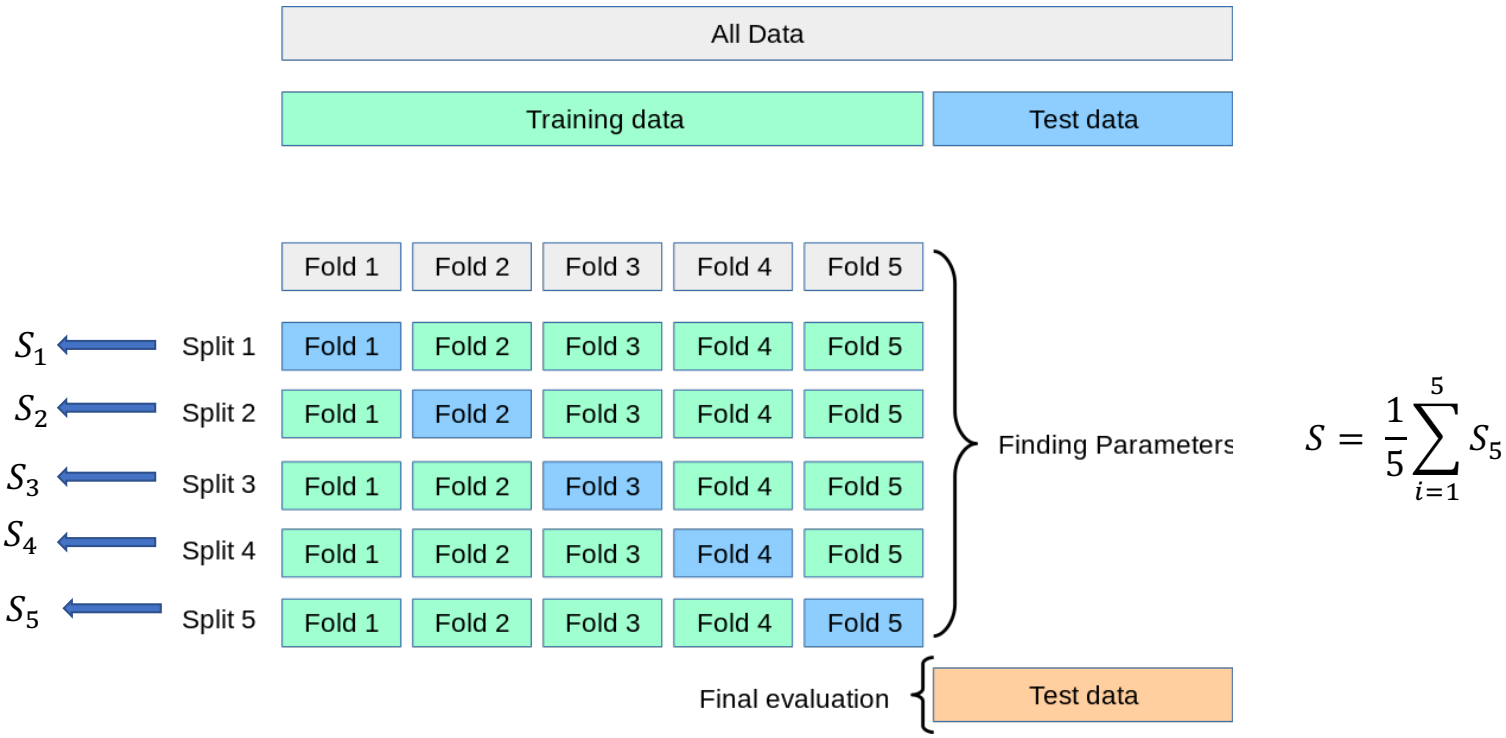




交叉验证 (Cross Validation)

不使用交叉验证（Cross Validation）

使用交叉验证（Cross Validation）



$$S = \frac{1}{5}\sum_{i=1}^{5} S_5$$

使用SKLearn内建的工具实现交叉验证调参

```
parameter_candidates = [
        {'C': [1, 10, 100, 1000], 'kernel': ['linear']},
        {'C': [1, 10, 100, 1000], 'gamma': [0.001, 0.0001], 'kernel': ['rbf']}
]
clf=GridSearchCV(estimator=svm.SVC(), param_grid=parameter_candidates, cv=5, n_jobs=-1)
clf.fit(X_train, y_train)

print('最佳模型的分数:', clf.best_score_)
print('最佳 C:',clf.best_estimator_.C)
print('最佳 Kernel:',clf.best_estimator_.kernel)
print(最佳 Gamma:',clf.best_estimator_.gamma)

best_model = clf.best_estimator_
best_model.score(X_test, y_test)
best_model.predict(X_test)
```

VC 维

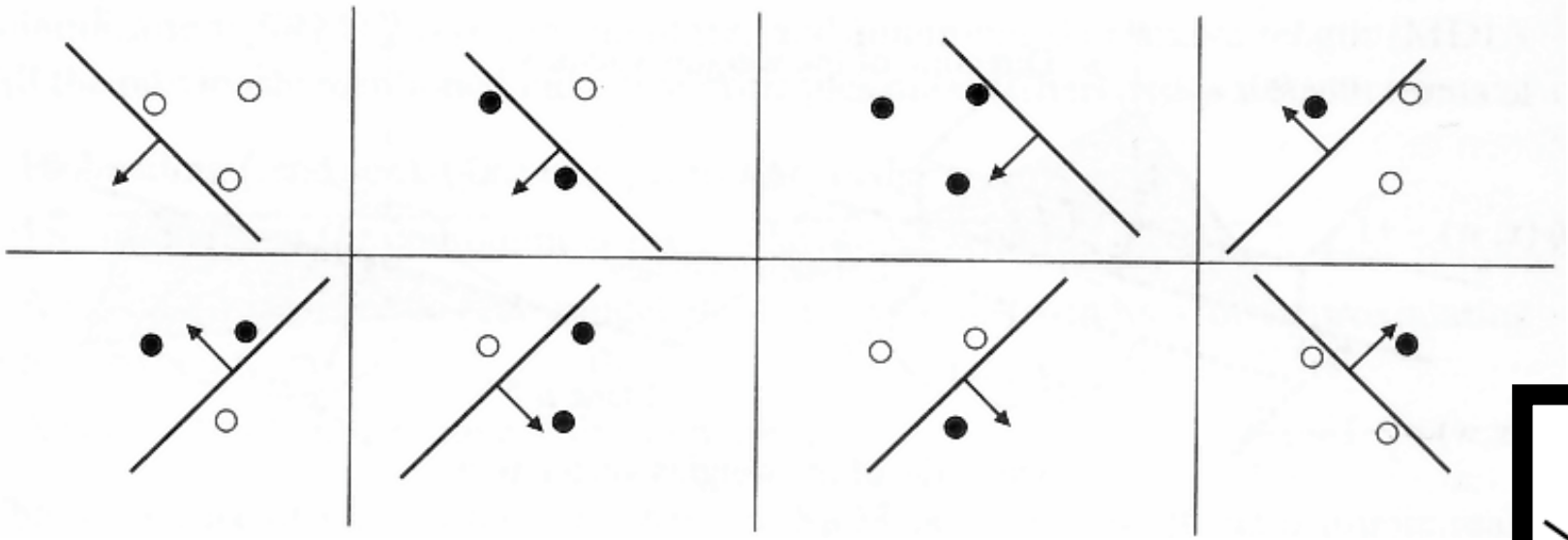
VC维理论

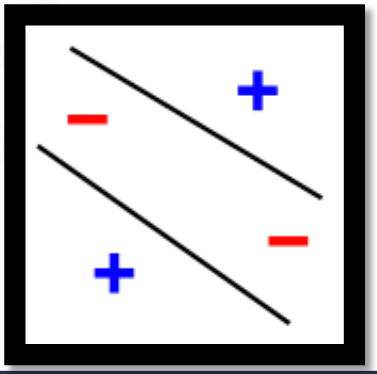
- 一个假设空间H的VC dimension，是这个H最多能够shatter掉的点的数量，记为dvc(H)。
- 假设空间可以看作模型的复杂度。
- shatter翻译成打散，指的是不管数据的分布如何，H都要把它区分开。
- ***“这个H最多能够shatter掉的点的数指的是无论数据的分布如何”***，这句话是指，不管数据是怎样分布的，H最多能区分多少个数据。我们可以想像，越是复杂的H能够区分的数据点就越多，VC维也就越大。

- 注意: 只要有一组数据点(个数为n)能被shatter, 就认为VC维度为n, 不要求所有的个数为n的数据点都能被shatter.

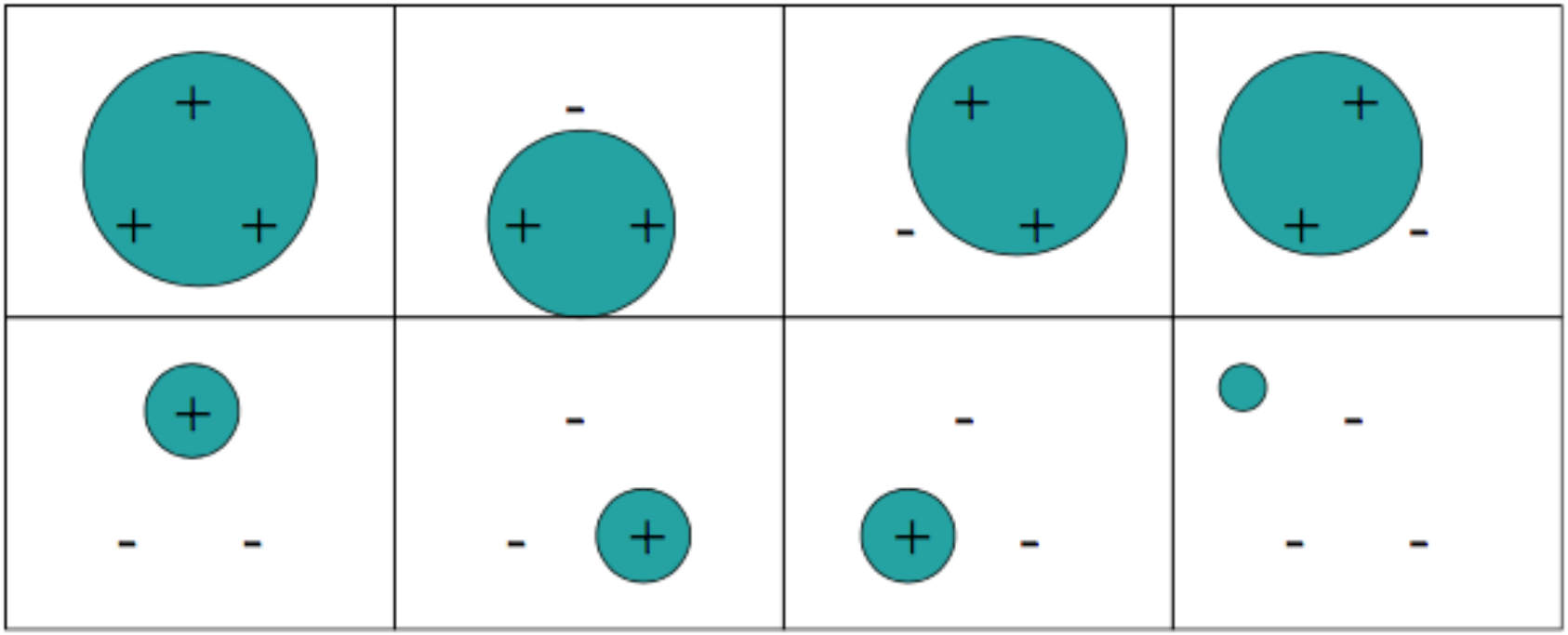
VC维实例：线性分类器的VC维为3

对于3个数据点，有8种赋值类别标签的方式 $2^3 = 8$，线性分类器总能把其正确分为两类，但是 4 个数据点就不行





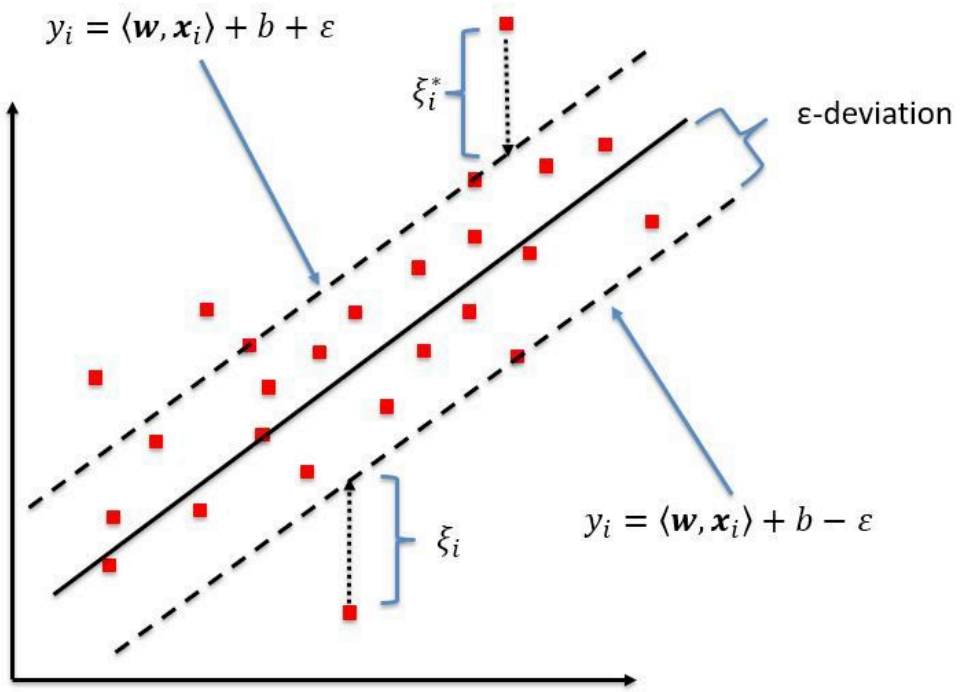
圆形框为分类器的实例，VC维为3



SVR 支持向量回归

参考: https://alex.smola.org/papers/2003/SmoSch03b.pdf

SVR



- Solution:

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

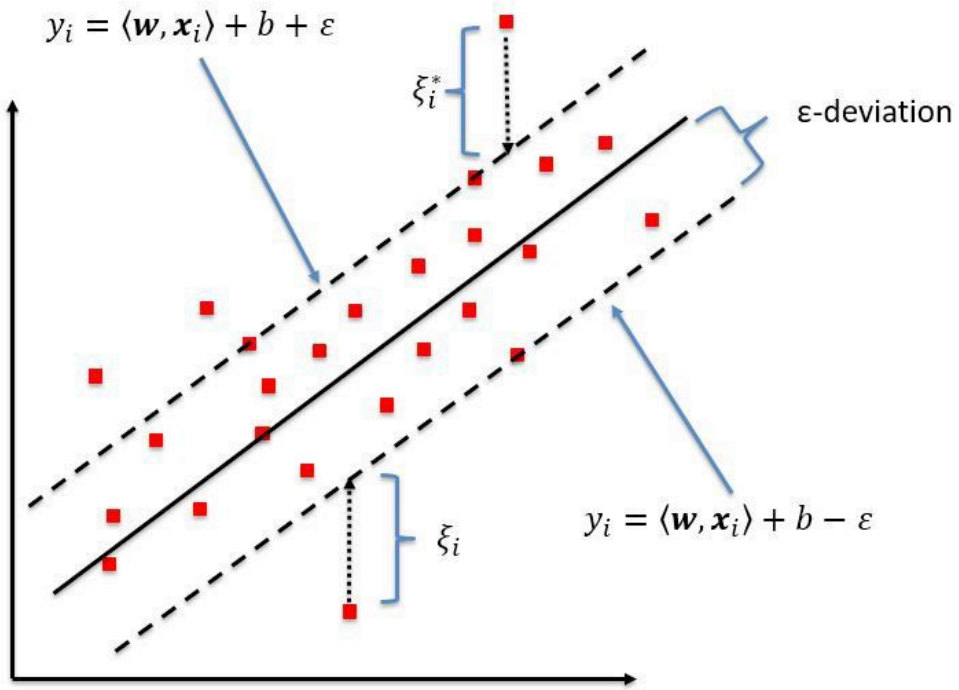
- Constraints:

$$y_i - wx_i - b \le \varepsilon$$

$$wx_i + b - y_i \le \varepsilon$$

$$y = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) \cdot \langle x_i, x \rangle + b$$

带松弛变量的SVR



- Minimize:

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\left(\xi_i + \xi_i^*\right)$$

- Constraints:

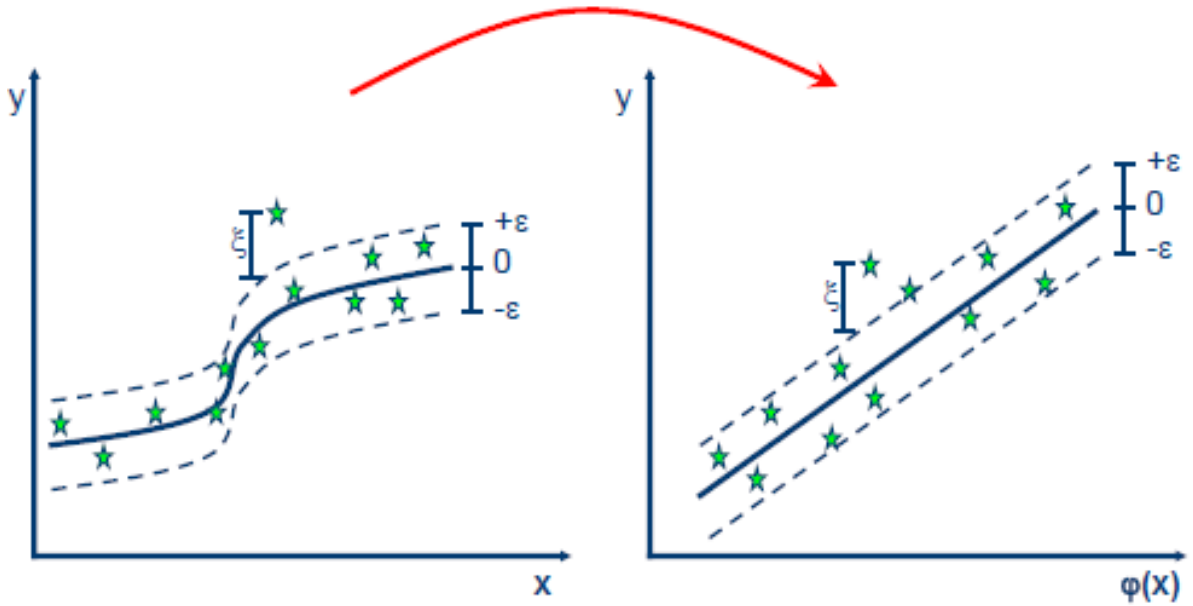
$$y_i - wx_i - b \le \varepsilon + \xi_i$$

$$wx_i + b - y_i \le \varepsilon + \xi_i^*$$

$$\xi_i, \xi_i^* \ge 0$$

$$y = \sum_{i=1}^{N}\left(\alpha_i - \alpha_i^*\right)\cdot\langle x_i, x\rangle + b$$

带松弛变量和核函数的SVR



$$y=\sum_{i=1}^{N}\left(\alpha_i-\alpha_i^*\right)\cdot\left\langle\varphi(x_i),\varphi(x)\right\rangle+b$$

$$y=\sum_{i=1}^{N}\left(\alpha_i-\alpha_i^*\right)\cdot K(x_i,x)+b$$

SVR 的数学推导

- 问题描述:
  - 给定 $\{(x_1, y_1), \ldots, (x_\ell, y_\ell)\} \subset \mathcal{X} \times \mathbb{R}$

    $$\mathcal{X} = \mathbb{R}^d$$

  - 求 $f(x) = \langle w, x \rangle + b \text{ with } w \in \mathcal{X}, b \in \mathbb{R}$

  - SVR模型

    $$\begin{aligned} &\text{minimize} && \tfrac{1}{2}\|w\|^2 \\ &\text{subject to} && \begin{cases} y_i - \langle w, x_i \rangle - b & \leq \quad \varepsilon \\ \langle w, x_i \rangle + b - y_i & \leq \quad \varepsilon \end{cases} \end{aligned}$$

    $$\|w\|^2 = \langle w, w \rangle$$

带松弛变量的SVR 的数学推导

- 问题描述:
  - 给定 $\{(x_1, y_1), \ldots, (x_\ell, y_\ell)\} \subset \mathcal{X} \times \mathbb{R}$

    $\mathcal{X} = \mathbb{R}^d$

  - 求 $f(x) = \langle w, x \rangle + b \text{ with } w \in \mathcal{X}, b \in \mathbb{R}$

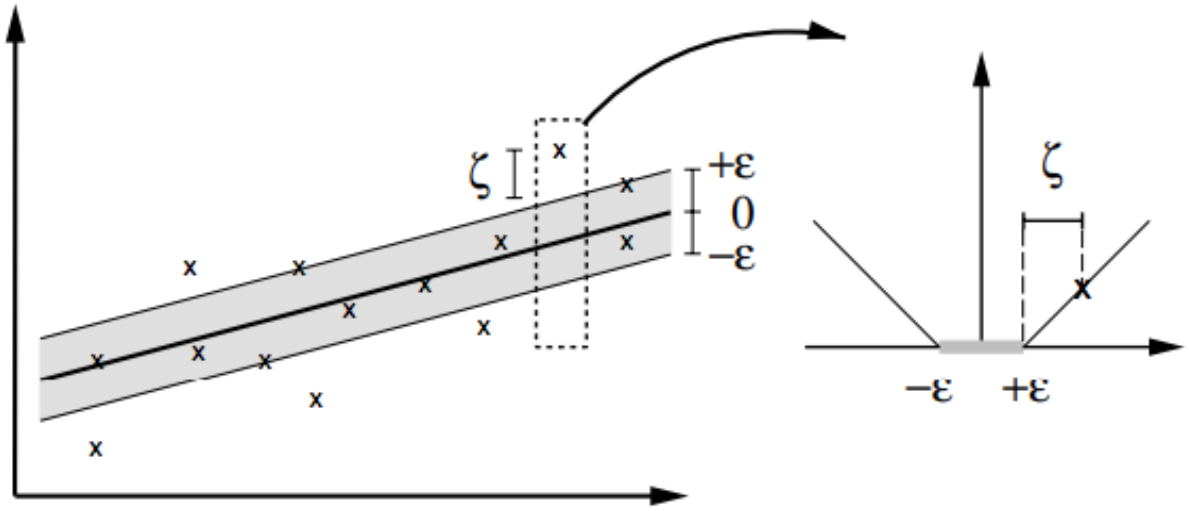
  - SVR模型 $C > 0$

$$\begin{aligned} \textbf{minimize} \quad & \tfrac{1}{2}\|w\|^2 + C \sum_{i=1}^{\ell} (\xi_i + \xi_i^*) \\ \textbf{subject to} \quad & \begin{cases} y_i - \langle w, x_i \rangle - b & \leq & \varepsilon + \xi_i \\ \langle w, x_i \rangle + b - y_i & \leq & \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* & \geq & 0 \end{cases} \end{aligned}$$

带松弛变量的SVR 的数学推导（续1）

- 损失函数 ε–insensitive loss function

$$|\xi|_\epsilon := \begin{cases} 0 & \text{if } |\xi| \le \epsilon \\ |\xi| - \epsilon & \text{otherwise.} \end{cases}$$



带松弛变量的SVR的dual对偶问题

$$\text{minimize} \quad \tfrac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*)$$

$$\text{subject to} \quad \begin{cases} y_i - \langle w, x_i\rangle - b & \leq \quad \varepsilon + \xi_i \\ \langle w, x_i\rangle + b - y_i & \leq \quad \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* & > \quad 0 \end{cases}$$

$$\alpha_i^{(*)}, \eta_i^{(*)} \geq 0.$$

$$L := \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*) - \sum_{i=1}^{\ell}(\eta_i\xi_i + \eta_i^*\xi_i^*)$$

$$- \sum_{i=1}^{\ell}\alpha_i(\varepsilon + \xi_i - y_i + \langle w, x_i\rangle + b)$$

$$- \sum_{i=1}^{\ell}\alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle w, x_i\rangle - b)$$

带松弛变量的SVR的dual对偶问题（续）

$$
\begin{aligned}
L \quad := \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*) - \sum_{i=1}^{\ell}(\eta_i\xi_i + \eta_i^*\xi_i^*) \\
& - \sum_{i=1}^{\ell}\alpha_i(\varepsilon + \xi_i - y_i + \langle w, x_i\rangle + b) \\
& - \sum_{i=1}^{\ell}\alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle w, x_i\rangle - b)
\end{aligned}
$$

为使得 L 取得极值，对primal参数求导，值应该为0

$$
\begin{aligned}
\partial_b L &= \sum_{i=1}^{\ell}(\alpha_i^* - \alpha_i) &&= 0 \\
\partial_w L &= w - \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)x_i &&= 0 \\
\partial_{\xi_i^{(*)}} L &= C - \alpha_i^{(*)} - \eta_i^{(*)} &&= 0
\end{aligned}
$$

带松弛变量的SVR的dual对偶问题（续2）

$$
\begin{aligned}
L \quad := \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*) - \sum_{i=1}^{\ell}(\eta_i\xi_i + \eta_i^*\xi_i^*) \\
& - \sum_{i=1}^{\ell}\alpha_i(\varepsilon + \xi_i - y_i + \langle w, x_i\rangle + b) \\
& - \sum_{i=1}^{\ell}\alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle w, x_i\rangle - b)
\end{aligned}
$$

$$
\begin{aligned}
\partial_b L = \quad & \sum_{i=1}^{\ell}(\alpha_i^* - \alpha_i) = 0 \\
\partial_w L = \quad & w - \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)x_i = 0 \\
\partial_{\xi_i^{(*)}} L = \quad & C - \alpha_i^{(*)} - \eta_i^{(*)} = 0 \Rightarrow \eta_i^{(*)} = C - \alpha_i^{(*)}
\end{aligned}
$$

$$
\text{maximize} \quad \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{\ell}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle x_i, x_j\rangle \\ -\varepsilon\sum_{i=1}^{\ell}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{\ell} y_i(\alpha_i - \alpha_i^*) \end{cases}
$$

$$
\text{subject to} \quad \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*) = 0 \text{ and } \alpha_i, \alpha_i^* \in [0, C]
$$

$$
w = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)x_i
$$

$$
f(x) = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)\langle x_i, x\rangle + b
$$

带松弛变量的SVR的dual对偶问题（续3）

$$\text{maximize} \quad \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{\ell}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle x_i, x_j\rangle \\ -\varepsilon\sum_{i=1}^{\ell}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{\ell} y_i(\alpha_i - \alpha_i^*) \end{cases}$$

$$\text{subject to} \quad \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*) = 0 \text{ and } \alpha_i, \alpha_i^* \in [0, C]$$

$$f(x) = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)\langle x_i, x\rangle + b$$

$$\text{minimize} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*)$$

$$\text{subject to} \quad \begin{cases} y_i - \langle w, x_i\rangle - b & \leq & \varepsilon + \xi_i \\ \langle w, x_i\rangle + b - y_i & \leq & \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* & \geq & 0 \end{cases}$$

- 计算 b 的值, 使用KKT (Karush-Kuhn-Tucker) 条件: 在最优解的点, 对偶变量和约束的乘积为0

$$\begin{aligned} \alpha_i(\varepsilon + \xi_i - y_i + \langle w, x_i\rangle + b) &= 0 \\ \alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle w, x_i\rangle - b) &= 0 \\ (C - \alpha_i)\xi_i &= 0 \\ (C - \alpha_i^*)\xi_i^* &= 0. \end{aligned}$$

1.对应 $\alpha_i^{(*)} = C$ (不为0) 的那些点在边界之外

2. 不对存在一对对偶变量同时为非0情况

$$\alpha_i\alpha_i^* = 0,$$

带松弛变量的SVR的dual对偶问题（续4）

$$\begin{aligned}
\text{minimize} \quad & \tfrac{1}{2}\|w\|^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*) \\
\text{subject to} \quad & \begin{cases} y_i - \langle w, x_i\rangle - b & \leq \varepsilon + \xi_i \\ \langle w, x_i\rangle + b - y_i & \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* & \geq 0 \end{cases}
\end{aligned}$$

$$\begin{aligned}
\alpha_i(\varepsilon + \xi_i - y_i + \langle w, x_i\rangle + b) &= 0 \\
\alpha_i^*(\varepsilon + \xi_i^* + y_i - \langle w, x_i\rangle - b) &= 0 \\
(C - \alpha_i)\xi_i &= 0 \\
(C - \alpha_i^*)\xi_i^* &= 0.
\end{aligned}$$

1.对应 $\alpha_i^{(*)} = C$ (不为0) 的那些点在边界之外

2. 不对存在一对对偶变量同时为非0情况 $\alpha_i\alpha_i^* = 0$

$$\begin{aligned}
&\varepsilon - y_i + \langle w, x_i\rangle + b \geq 0 \quad \text{and} \quad \xi_i = 0 \quad \text{if} \quad \alpha_i < C \\
&\varepsilon - y_i + \langle w, x_i\rangle + b \leq 0 \qquad\qquad\quad \text{if} \quad \alpha_i > 0
\end{aligned}$$

$$\begin{aligned}
&\max\{-\varepsilon + y_i - \langle w, x_i\rangle | \alpha_i < C \text{ or } \alpha_i^* > 0\} \leq b \leq \\
&\min\{-\varepsilon + y_i - \langle w, x_i\rangle | \alpha_i > 0 \text{ or } \alpha_i^* < C\}
\end{aligned}$$

只有在边界之外的数据点，代入如下公式才成立. 这些点被称为支持向量，支持向量唯一决定了回归曲线/面.

$$|f(x_i) - y_i| \geq \varepsilon$$

带松弛变量和核函数的SVR的dual对偶问题

$$\begin{aligned}
&\text{maximize} \quad \begin{cases} -\frac{1}{2}\sum_{i,j=1}^{\ell}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)k(x_i, x_j) \\ -\varepsilon\sum_{i=1}^{\ell}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{\ell} y_i(\alpha_i - \alpha_i^*) \end{cases} \\
&\text{subject to} \quad \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*) = 0 \text{ and } \alpha_i, \alpha_i^* \in [0, C]
\end{aligned}$$

$$w = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)\Phi(x_i)$$

$$f(x) = \sum_{i=1}^{\ell}(\alpha_i - \alpha_i^*)k(x_i, x) + b.$$

SVR Primal V.S. Dual

- Primal

$$\min \frac{1}{2}\| \mathbf{w} \|^2 + C\sum_{i=1}^{m}(\xi_i + \xi_i^*)$$

$$s.t.\begin{cases} y_i - (\mathbf{w}\cdot\varphi(\mathbf{x}_i)) - b \le \varepsilon + \xi_i \\ (\mathbf{w}\cdot\varphi(\mathbf{x}_i)) + b - y_i \le \varepsilon + \xi_i^* \\ \xi_i\ ,\xi_i^* \ge 0, i = 1,\ldots,m \end{cases}$$

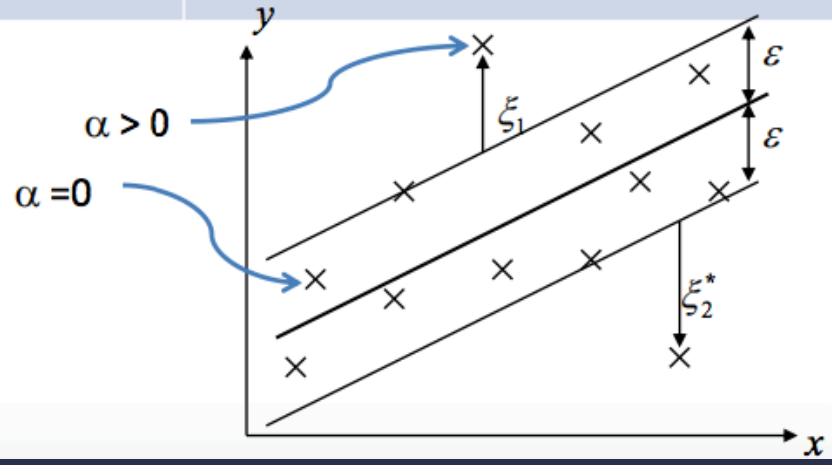
- Dual

K(xi, xj)

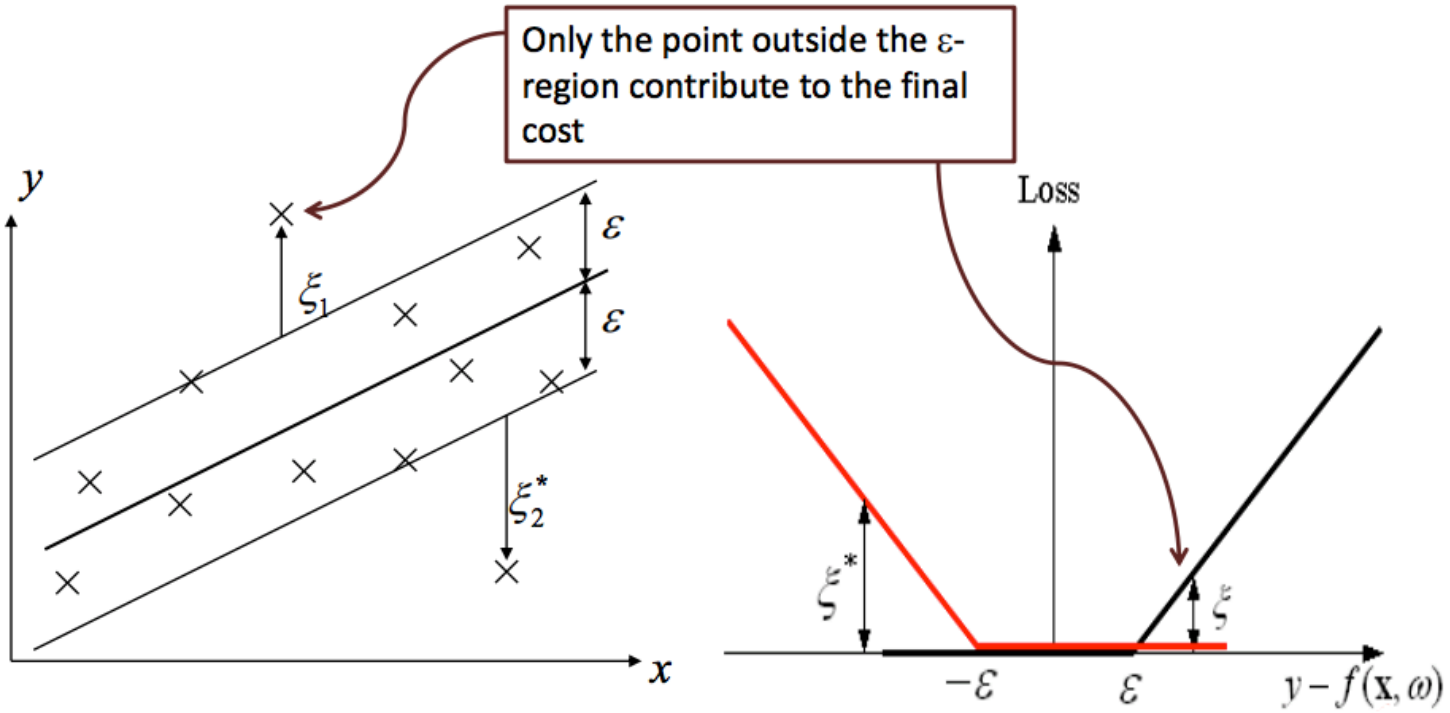
$$\max\begin{cases} \frac{1}{2}\sum_{i,j=1}^{m}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle\varphi(\mathbf{x}_i),\varphi(\mathbf{x}_j)\rangle \\ -\varepsilon\sum_{i=1}^{m}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{m} y_i(\alpha_i - \alpha_i^*) \end{cases}$$

$$s.t.\sum_{i=1}^{m}(\alpha_i - \alpha_i^*) = 0;\ 0 \le \alpha_i, \alpha_i^* \le C$$

Primal variables: w for each feature dim	Dual variables: α, α* for each data point
Complexity: the dim of the input space	Complexity: Number of support vectors



使用损失函数理解 SVR



$$L_\varepsilon(y, f(\mathbf{x}, \omega)) = \max(|\, y - f(\mathbf{x}, \omega)\,| - \varepsilon, 0)$$

贪心科技
GREEDY TECHNOLOGY

贪心科技　让每个人享受个性化教育服务

THANKS

贪心学院讲师：袁源