第3章

支持向量机与核学习

Support Vector Machine & Kernel Learning

向 世 明

smxiang@nlpr.ia.ac.cn

http://www.escience.cn/people/smxiang/index.html

中科院自动化研究所 模式识别国家重点实验室

助教： 方深 (shen.fang@nlpr.ia.ac.cn)

University of Chinese Academy of Sciences

内容提要

- 感知器准则
- 函数间隔、几何间隔、间隔最大化
- 支持向量机
- 支持向量机回归
- 支持向量机排序
- 核技巧
- KSVM
- KPCA、KLDA、关于核化的一般性理论 （不讲）

3.1 感知器准则

- **线性判别函数：**

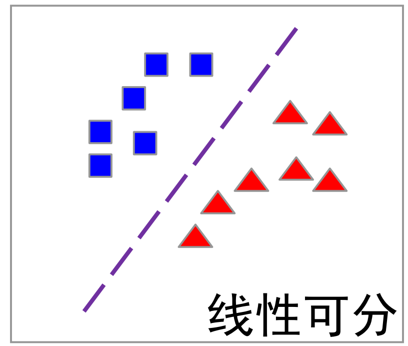
$$g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b, \quad \mathbf{x} \in R^d$$

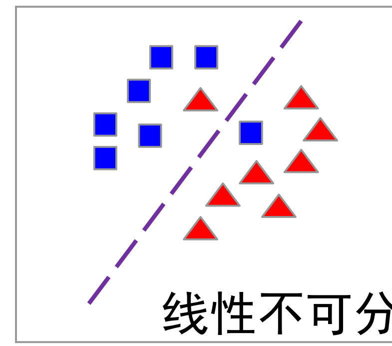
权重向量　　偏移（阈值）

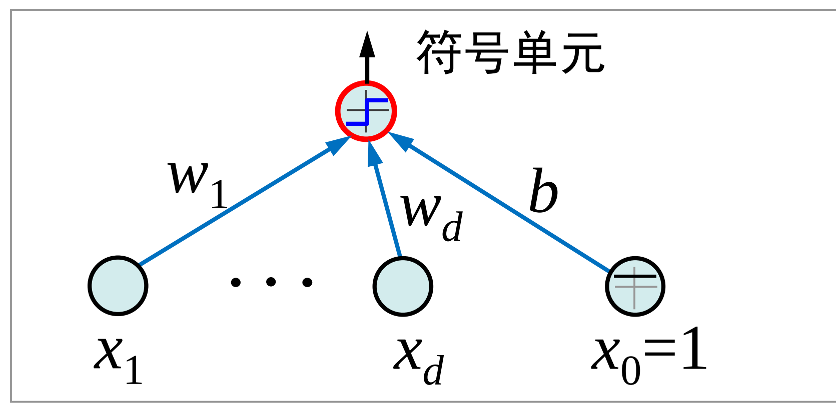
- **两类分类决策规则：**

$$\begin{cases} \mathbf{x} \in \omega_1, & \text{if } g(\mathbf{x}) > 0 \\ \mathbf{x} \in \omega_2, & \text{if } g(\mathbf{x}) < 0 \\ \text{uncertain}, & \text{if } g(\mathbf{x}) = 0 \end{cases}$$

给定n个训练样本$(\mathbf{x}_1, y_1)$，$(\mathbf{x}_2, y_2)$, …，$(\mathbf{x}_n, y_n)$，其中$\mathbf{x}_i \in R^d$，i=1, 2, …, n 为 d 维空间中的样本特征，$y_i \in \{+1,-1\}$为其对应的类别标签。







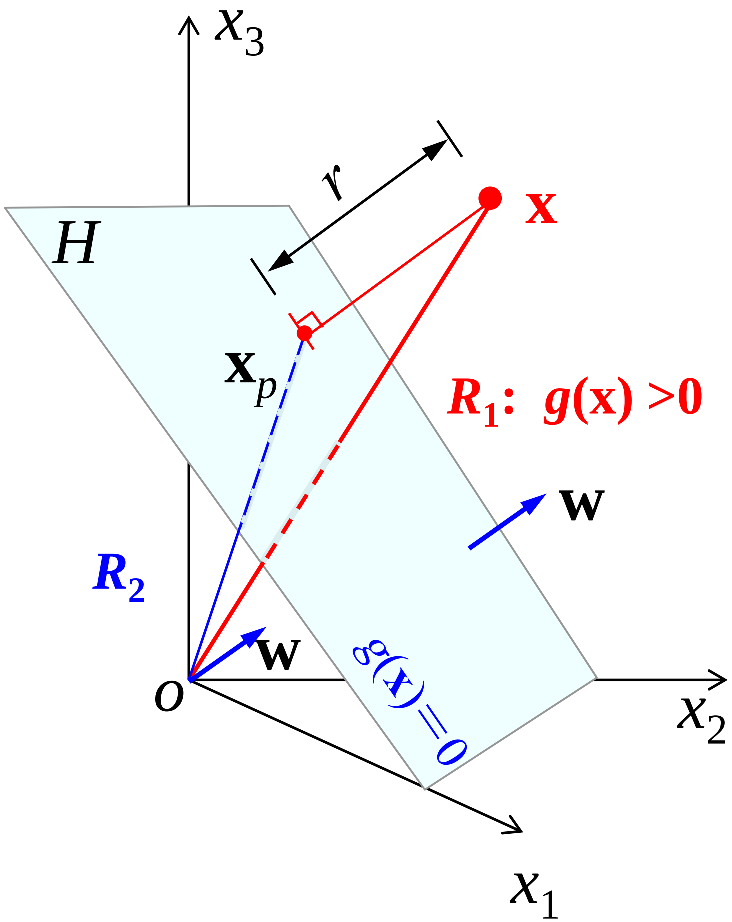
线性分类器（神经网络）

3.1 感知器准则

- 两类情形的决策面
  - 决策面方程：$g(\mathbf{x})=\mathbf{w}^T\mathbf{x}+b=0$
  - 对于任意样本$\mathbf{x}$，将其向决策面内投影，并写成两个向量之和：

$$\mathbf{x}=\mathbf{x}_p+r\frac{\mathbf{w}}{\|\mathbf{w}\|}$$

其中，$\mathbf{x}_p$为 $\mathbf{x}$ 在超平面 H上的投影，r 为点 $\mathbf{x}$ 到超平面 H的代数距离。如果 $\mathbf{x}$ 在超平面正侧，则 $r>0$; 反之 $r<0$ 。



3.1 感知器准则

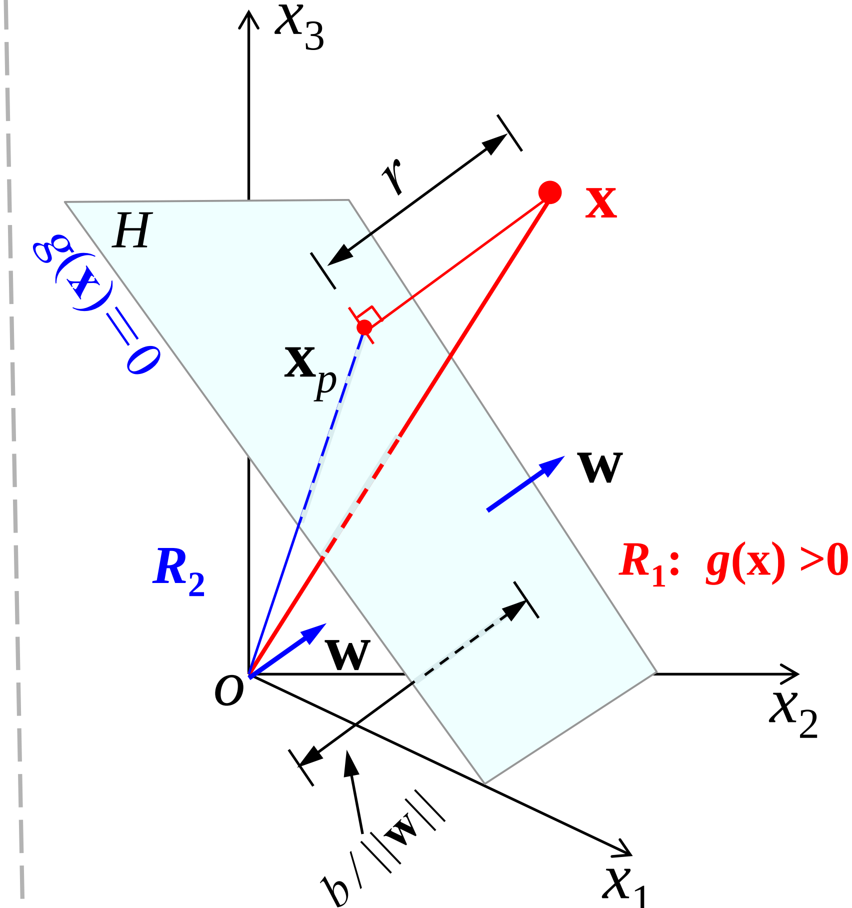
- 两类情形的决策面
  - 注意 $g(\mathbf{x}_p) = 0$，于是有：

$$g(\mathbf{x}) = \mathbf{w}^T \left(\mathbf{x}_p + r \frac{\mathbf{w}}{\|\mathbf{w}\|} \right) + b$$

$$= r \|\mathbf{w}\|$$

$$\Rightarrow \quad r = \frac{g(\mathbf{x})}{\|\mathbf{w}\|} \quad \text{（符号距离）}$$

此外，可得坐标原点到超平面的距离为：$b / \|\mathbf{w}\|$



3.1 感知器准则

- 感知器准则
  - **损失函数的一个自然选择是被误分样本点的总数**。但它不是 $\mathbf{w}$ 和 b 的连续可导函数，难以优化。
  - 因此，转而最小化误分点到分类超平面的距离。
  - 对于误分点$\mathbf{x}_i$而言，总有：$-y_i(\mathbf{w}^T\mathbf{x}_i + b) > 0$
  - 这样有如下目标函数：

$$\min_{\mathbf{w},b} \quad -\frac{1}{\|\mathbf{w}\|}\sum_{\mathbf{x}_i \in M} y_i(\mathbf{w}^T\mathbf{x}_i + b)$$

（即误分点到$g(\mathbf{x})=0$的距离）

  - 不考虑系数 $-1/\|\mathbf{w}\|$，可得感知器学习的损失函数：

$$\min_{\mathbf{w},b} \quad -\sum_{\mathbf{x}_i \in M} y_i(\mathbf{w}^T\mathbf{x}_i + b)$$

3.1 感知器准则

- **算法步骤**

Batch Perceptron一基本算法　(可分情形）

begin initialize: $\mathbf{w}$, b, η, certain θ (small value), t=0

　do $t \leftarrow t+1$

　　找出当前所有错分点即（$y_i\,(\mathbf{w}^T\mathbf{x}_i + b) \le 0$），记录于$M_t$：

　　$\mathbf{w} = \mathbf{w} + \eta \sum_{\mathbf{x}_i \in M_t} y_i \mathbf{x}_i$

　　$b = b + \eta \sum_{\mathbf{x}_i \in M_t} y_i$

　until M_t={}, or $\eta \sum_{\mathbf{x}_i \in M_t} y_i < \theta$　　// 一个较松的停止条件

　return $\mathbf{w}$, b

end

3.1 感知器准则

- 如下问题的解有不同之处吗？在什么情况下两者的优化结果是一样的？

原问题（1）(几何间隔)： $$\min_{\mathbf{w},b} \; -\frac{1}{\|\mathbf{w}\|}\sum_{\mathbf{x}_i \in M}\left(y_i(\mathbf{w}^T\mathbf{x}_i + b)\right)$$

新问题（2）(函数间隔)： $$\min_{\mathbf{w},b} \; -\sum_{\mathbf{x}_i \in M}\left(y_i(\mathbf{w}^T\mathbf{x}_i + b)\right)$$

结论：在线性可分情形下是一样的，在线性不可以分情形下则不一样。

3.1 感知器准则

- 优缺点
  - 感知器准则计算简单，对可分情形在有限步迭代后一定收敛。
  - 感知器准则每次采用校度下降方法，但每一次校度下降所对应的目标函数均不同，导致算法起伏不定！
  - 能不能有更好地办法来解决这一问题？

3.2 函数间隔、几何间隔、间隔最大化

- 函数间隔
  - 一个点**距离分类超平面的远近**可以表示分类预测的确信程度。
  - 样本点函数间隔：

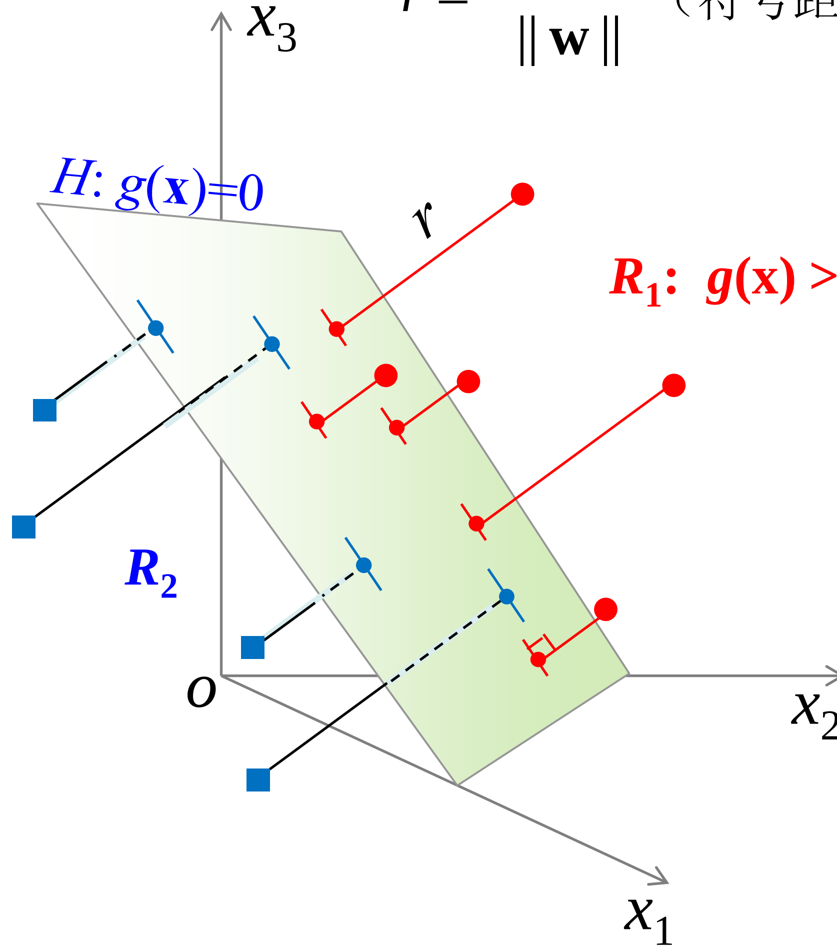
    $$\hat{r}_i = y_i(\mathbf{w}^T\mathbf{x}_i + b)$$

  - 超平面 H 关于训练集 T 的函数间隔：

    $$\hat{r} = \min\{\hat{r}_i\}_{i=1}^{n}$$

  - 仅仅极小化函数间隔是不足够的！

$$r = \frac{\mathbf{w}^T\mathbf{x} + b}{\|\mathbf{w}\|}$$（符号距离）



3.2 函数间隔、几何间隔、间隔最大化

- 几何间隔：
  - 样本点的几何间隔： $r_i = \frac{1}{\|\mathbf{w}\|} y_i(\mathbf{w}^T \mathbf{x}_i + b)$

- 超平面H关于训练集T的几何间隔：$r = \min \{r_i\}_{i=1}^{n}$
- 函数间隔与几何间隔的关系：

$$r_i = \frac{\hat{r}_i}{\|\mathbf{w}\|}, \quad r = \frac{\hat{r}}{\|\mathbf{w}\|}$$

可见：几何间隔不会随着w和b的改变而改变。

3.2 函数间隔、几何间隔、间隔最大化

- 间隔最大化：

最大化几何间隔

$$\max_{\mathbf{w},b} \quad r, \quad s.t. \quad \frac{1}{\|\mathbf{w}\|} y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq r, \quad i = 1, 2, \ldots, n$$

$$r_i = \frac{\hat{r}_i}{\|\mathbf{w}\|}, \quad r = \frac{\hat{r}}{\|\mathbf{w}\|}$$

$$\max_{\mathbf{w},b} \quad \frac{\hat{r}}{\|\mathbf{w}\|}, \quad s.t. \quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq \hat{r}, \quad i = 1, 2, \ldots, n$$

(令：$\hat{r} = 1$)

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\|\mathbf{w}\|^2, \quad s.t. \quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, \quad i = 1, 2, \ldots, n$$

3.2 函数间隔、几何间隔、间隔最大化

- 间隔最大化：

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

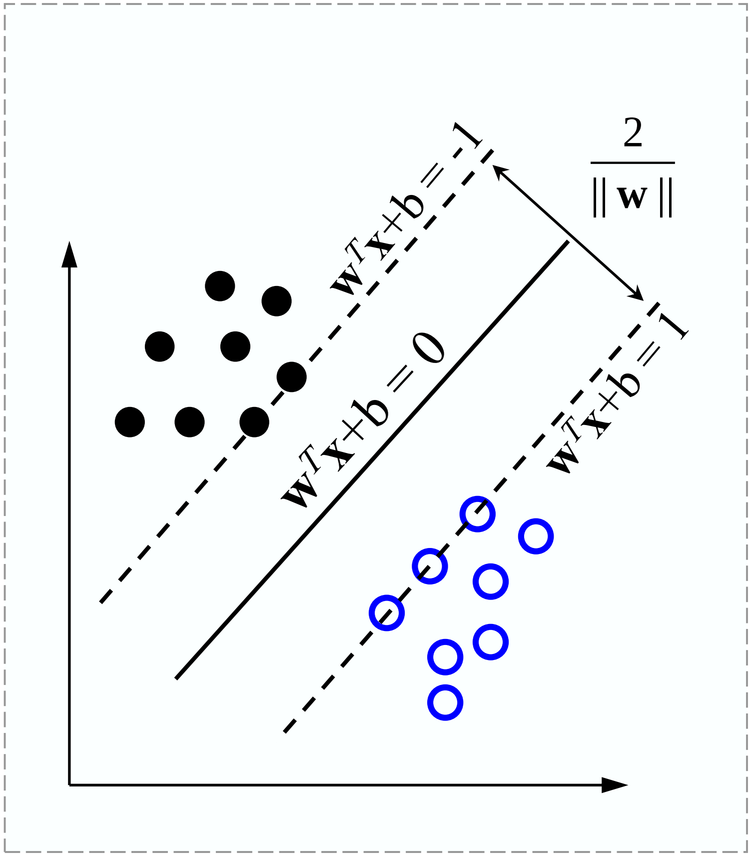
$$s.t. \quad y_i(\mathbf{w}^T\mathbf{x}_i+b)\geq 1,$$

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

$$\text{distance}(plane:\mathbf{w}^T\mathbf{x}_i+b=+1, plane:\mathbf{w}^T\mathbf{x}_i+b=-1)$$

$$=\frac{1}{\|\mathbf{w}\|}-\frac{-1}{\|\mathbf{w}\|}$$

$$=\frac{2}{\|\mathbf{w}\|}$$

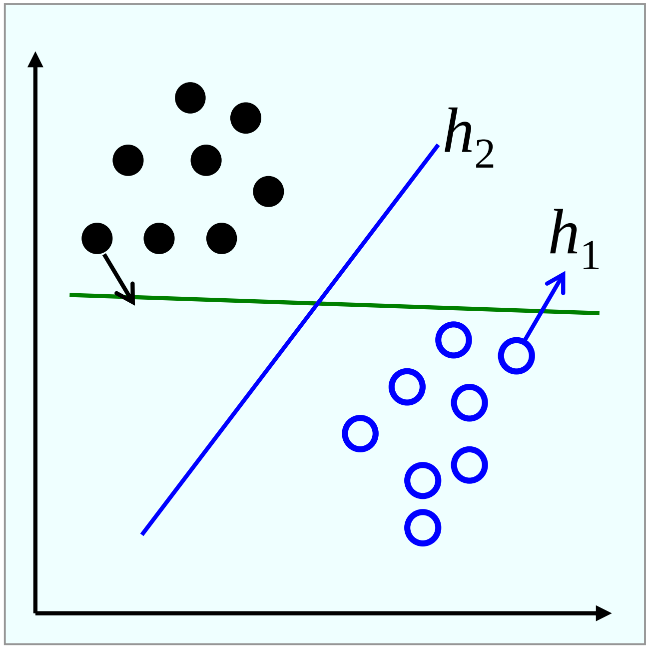


3.2 函数间隔、几何间隔、间隔最大化

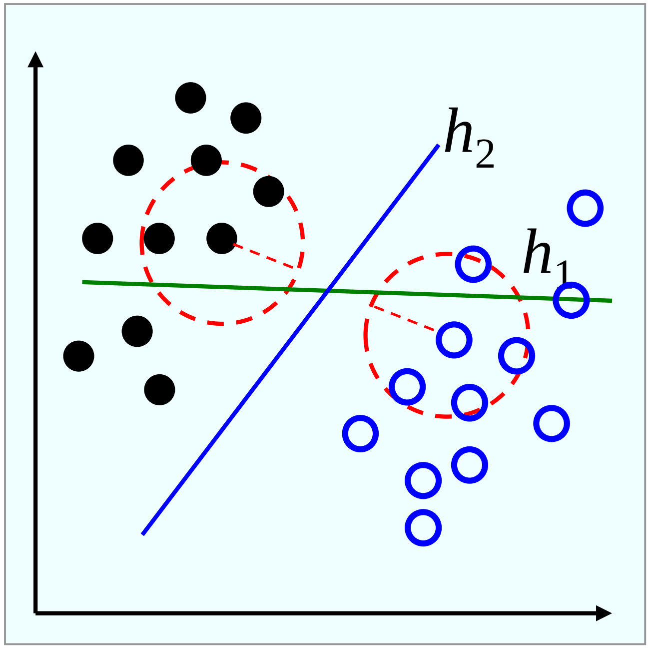
- 感知器准则与支持向量机
  - 感知器准则考虑错分样本，其损失函数为错误分类样本的函数间隔最小。
  - 支持向量机考虑所有样本，其损失函数为正确分类样本的函数间隔最大。
  - 单纯地从最优化的角度看，感知器规则具有**平凡解**。
  - 支持向量机不存在平凡解。

3.3 支持向量机

- **从分类器构造的直观角度**



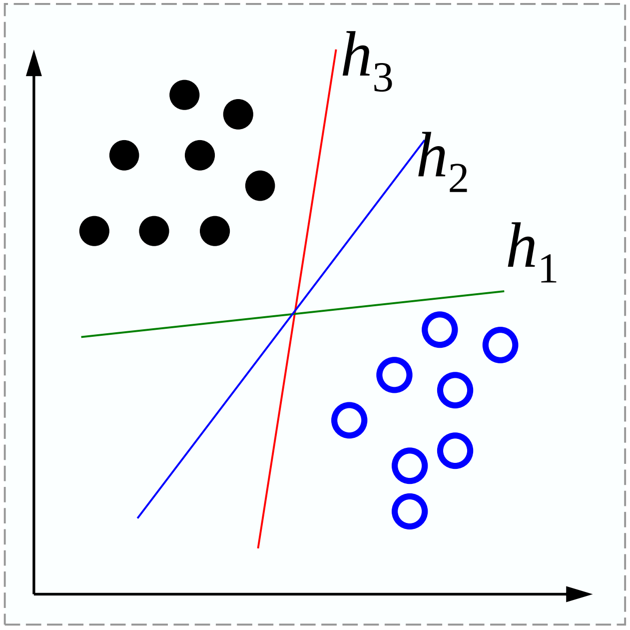
对局部扰动的稳定性



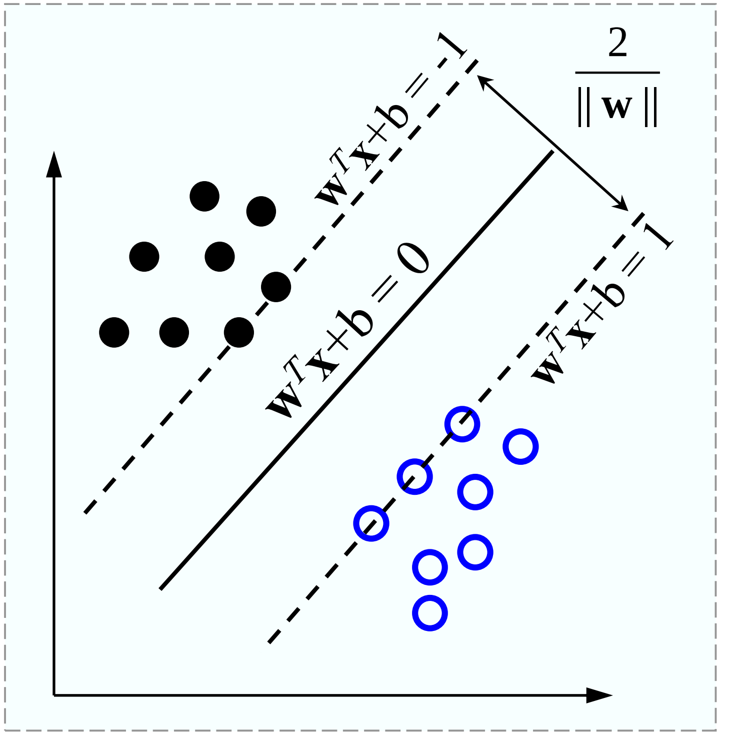
对未见样本的泛化能力

3.3 支持向量机

- **从分类器构造的直观角度**



类间数据点间隔最大

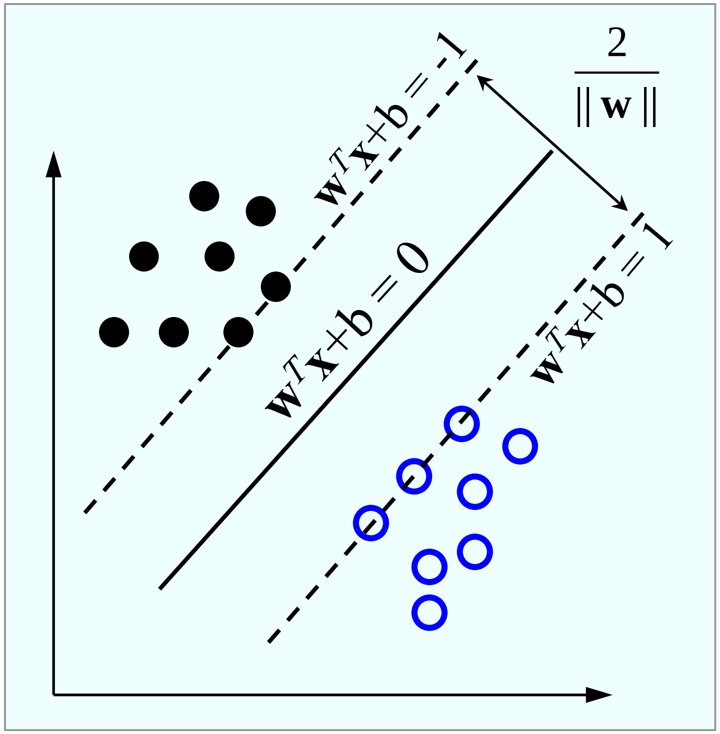


最大间隔分类超平面

3.3 支持向量机

- **学习模型**

线性可分情形



给定训练集：

$T=\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_n,y_n)\}, y_i\in\{+1, -1\}$

任务：估计最大间隔分类超平面

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

$$s.t. \quad y_i(\mathbf{w}^T\mathbf{x}_i+b)-1\geq 0,$$

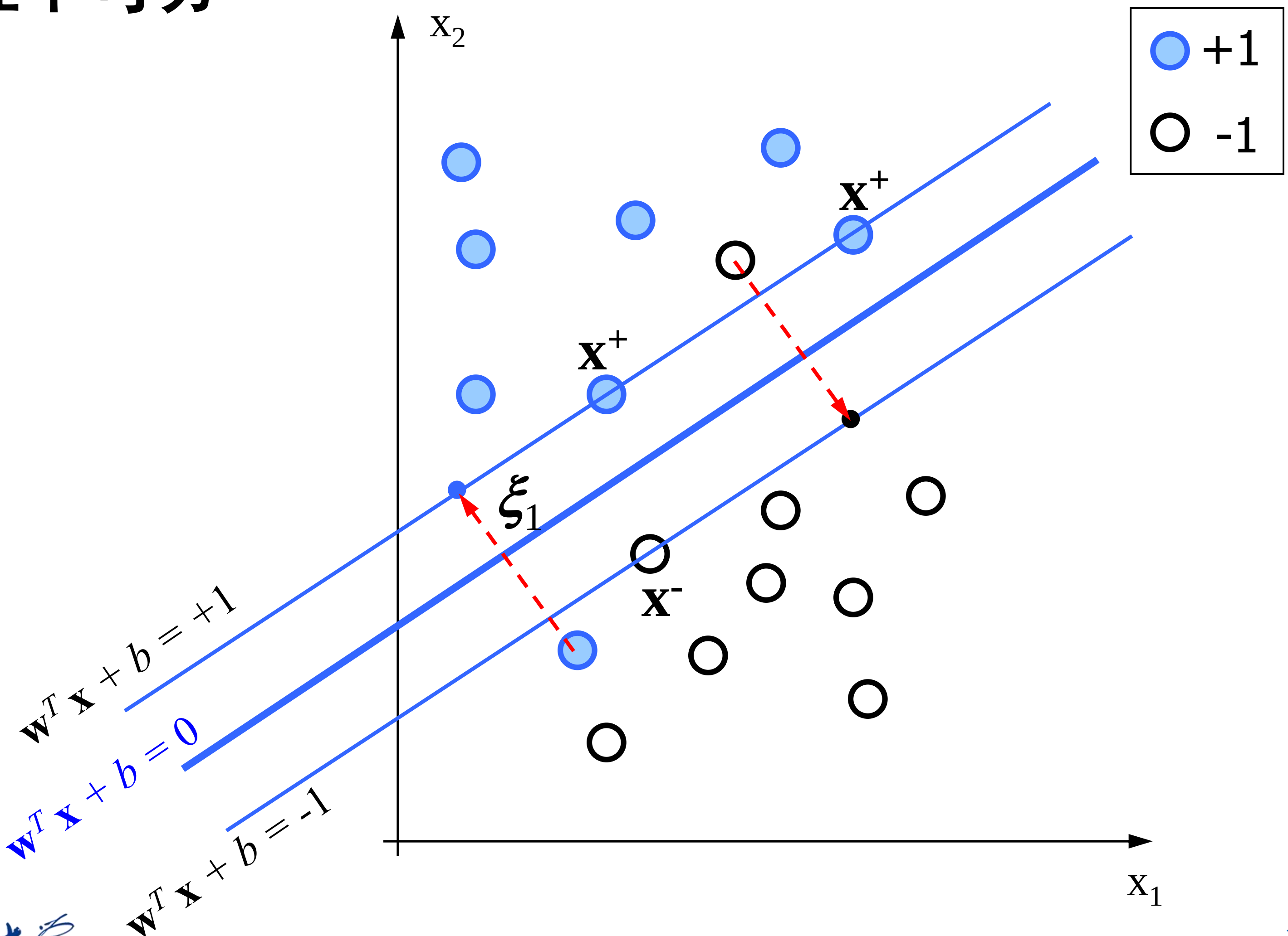
$$i=1,2,\cdots,n$$

分类超平面： $\mathbf{w}^T\mathbf{x}+b=0$

分类决策函数： $f(\mathbf{x})=sign(\mathbf{w}^T\mathbf{x}+b)$

3.3 支持向量机

- 线性不可分



3.3 支持向量机

- **硬间隔与软间隔**

线性可分情况：

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 \longleftarrow \text{硬间隔}$$

对于线性不可分情况，对约束条件引入松弛变量，允许有**少量**样本落在两类分类间隔中间：

$$y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0 \longleftarrow \text{软间隔}$$

$$\boxed{\min \quad \sum_{i=1}^{n} \xi_i}$$

3.3 支持向量机

- **线性不可分-学习模型**

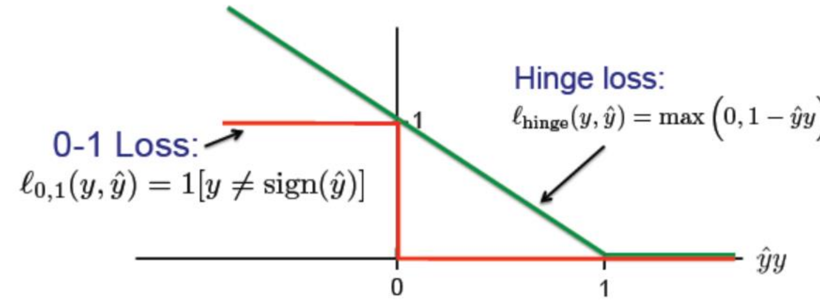
体现了表达能力　　体现了经验风险

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

$$\begin{aligned} s.t. \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \\ & i = 1, 2, \cdots, n \end{aligned}$$

目标函数第一项表示使**margin**尽量大，第二项表示使误差分类点的个数尽量小。

3.3 支持向量机



Hinge loss upper bounds 0/1 loss!

It is the tightest convex upper bound on the 0/1 loss

- **软间隔最大化：**
  - More robust for outliers

$$
\begin{aligned}
\min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\
s.t. \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq \underline{1-\xi_i}, \\
& \xi_i \geq 0, \\
& i = 1, 2, \cdots, n
\end{aligned}
\quad \Longleftrightarrow \quad
\min_{\mathbf{w},b} \sum_{i=1}^{n}\underline{[1-y_i(\mathbf{w}^T\mathbf{x}_i + b)]}_+ + \lambda\|\mathbf{w}\|^2
$$

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

合页损失函数

Hinge loss function!

3.3 支持向量机

- **对偶算法（线性可分情形）**
  - ✓ 对偶算法往往容易求解
  - ✓ 对偶算法可以推广到核学习

（拉格朗日对偶性）⇨

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\|\mathbf{w}\|^2$$
$$s.t. \quad y_i(\mathbf{w}^T\mathbf{x}_i+b)-1\geq 0, \quad i=1,2,\cdots,n$$

原始问题

$$\max_{\boldsymbol{\alpha}} \min_{\mathbf{w},b} \quad L(\mathbf{w},b,\boldsymbol{\alpha})$$
$$s.t. \quad \alpha_i \geq 0, \quad i=1,2,\ldots,n$$

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

对偶问题

3.3 支持向量机

- 对偶问题求解
  - (1) 求 $\min_{\mathbf{w},b} L(\mathbf{w},b,\boldsymbol{\alpha})$

$$\nabla_{\mathbf{w}} L(\mathbf{w},b,\boldsymbol{\alpha}) = \mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = \mathbf{0} \quad \Rightarrow \quad \mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i$$

$$\nabla_{b} L(\mathbf{w},b,\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i y_i = 0 \quad \Rightarrow \quad \sum_{i=1}^{n} \alpha_i y_i = 0$$

$$\min_{\mathbf{w},b} L(\mathbf{w},b,\boldsymbol{\alpha}) = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \underline{(\mathbf{x}_i \cdot \mathbf{x}_j)} + \sum_{i=1}^{n} \alpha_i$$

3.3 支持向量机

– (2) 求对偶问题，即求 $\min\limits_{\mathbf{w},b} L(\mathbf{w},b,\boldsymbol{\alpha})$ 对 $\boldsymbol{\alpha}$ 的极大

$$\max_{\boldsymbol{\alpha}} \quad -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i\cdot\mathbf{x}_j)+\sum_{i=1}^{n}\alpha_i$$

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

$$\alpha_i \ge 0, \quad i=1,2,\dots,n$$

$$\Downarrow$$

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i\cdot\mathbf{x}_j)-\sum_{i=1}^{n}\alpha_i$$

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

$$\alpha_i \ge 0, \quad i=1,2,\dots,n$$

3.3 支持向量机

- **定理1**

  - 设 $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_1^*, \cdots, \alpha_n^*]^T \in R^n$ 是对偶问题的解，则至少存在一个下标 j，使得$\alpha_j^* > 0$，可按下式求得原始问题的最优解：

$$\mathbf{w}^* = \sum_{i=1}^{n} \alpha_i^* y_i \mathbf{x}_i, \quad b^* = y_j - \sum_{i=1}^{n} \alpha_i^* y_i (\mathbf{x}_i \cdot \mathbf{x}_j)$$

- **分类超平面：** $\mathbf{w}^* \cdot \mathbf{x} + b^* = 0$
- **分类决策函数：** $f(\mathbf{x}) = sign(\mathbf{w}^* \cdot \mathbf{x} + b^*) = sign\left(\sum_{i=1}^{n} \alpha_i^* y_i (\mathbf{x} \cdot \mathbf{x}_i) + b^*\right)$

结论： 对线性可分情形 最优解 b^* 是唯一的。

3.3 支持向量机

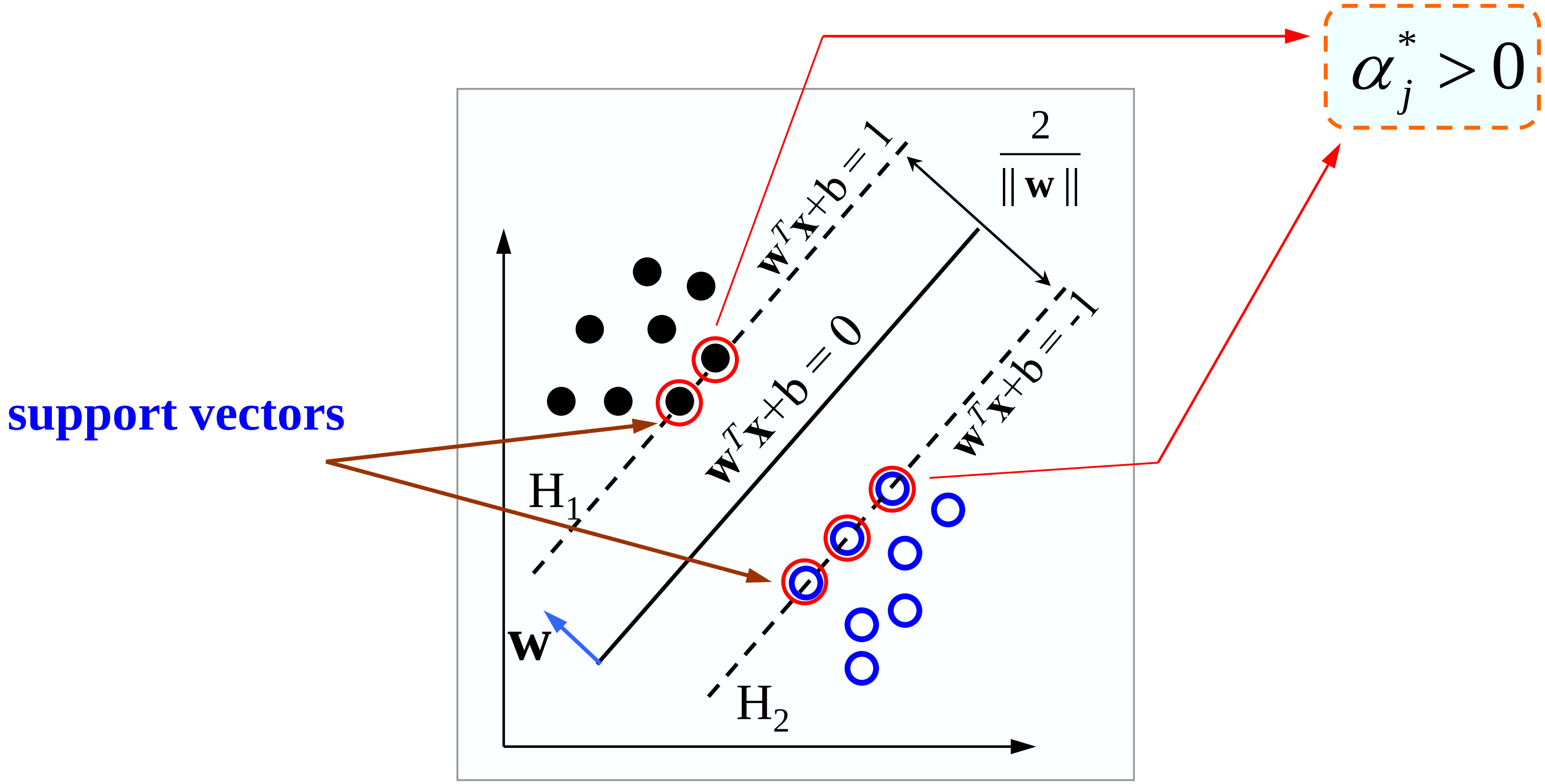
- **KKT条件：**

$$\left.\begin{aligned}
&\alpha_i^* \geq 0 \\
&y_i(\mathbf{w}^{*T}\mathbf{x}_i + b^*) - 1 \geq 0 \\
&\boxed{\alpha_i^*(y_i(\mathbf{w}^{*T}\mathbf{x}_i + b^*) - 1) = 0}
\end{aligned}\right\} \text{KKT条件！}$$

$$\alpha_i^* = 0, \quad y_i(\mathbf{w}^{*T}\mathbf{x}_i + b^*) \geq 1$$

$$\boxed{\alpha_i^* > 0, \quad y_i(\mathbf{w}^{*T}\mathbf{x}_i + b^*) = 1} \leftarrow \boxed{\text{支持向量}}$$

- 使等式成立的点为**支持向量**：$y_i(\mathbf{w}^T\mathbf{x}_i + b) = 1$

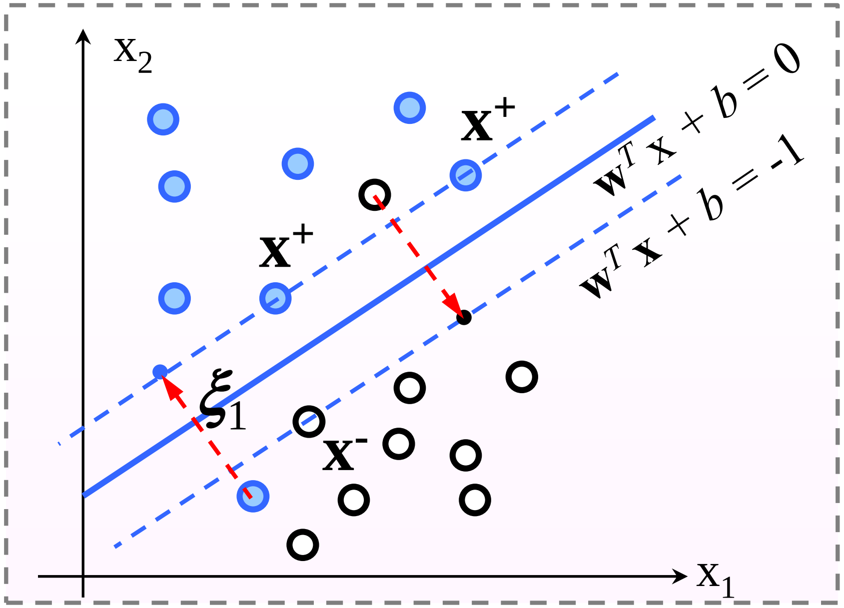


所有样本中，“支持向量”到分类面的几何距离最小。

软间隔最大化:

$$
\begin{aligned}
\min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\
s.t. \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \\
& \xi_i \geq 0, \\
& i = 1,2,\cdots,n
\end{aligned}
$$

原始问题



（广义拉格朗日函数）

$$L(\mathbf{w},b,\xi,\boldsymbol{\alpha},\mu) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\alpha_i y_i(\mathbf{w}^T\mathbf{x}_i + b + \xi_i) + \sum_{i=1}^{n}\alpha_i - \sum_{i=1}^{n}\mu_i\xi_i$$

拉格朗日对偶

$$\max_{\substack{\alpha\geq 0\\ \mu\geq 0}} \min_{w,b,\xi} L(w,b,\xi,\alpha,\mu)$$

$$
\begin{aligned}
\min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i \cdot \mathbf{x}_j) - \sum_{i=1}^{n}\alpha_i \\
s.t. \quad & \sum_{i=1}^{n}\alpha_i y_i = 0; \\
& 0 \leq \alpha_i \leq C, \quad i = 1,2,\ldots,n
\end{aligned}
$$

对偶问题

3.3 支持向量机

- **定理2**
  - 设 $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_1^*, \cdots, \alpha_n^*]^T \in R^n$ 是对偶问题的解，则至少存在一个下标 j，有 $0 < \alpha_j^* < C$，可按下式求得原始问题的最优解：

$$\mathbf{w}^* = \sum_{i=1}^{n} \alpha_i^* y_i \mathbf{x}_i, \quad b^* = y_j - \sum_{i=1}^{n} \alpha_i^* y_i (\mathbf{x}_i \cdot \mathbf{x}_j)$$

- 分类超平面： $\mathbf{w}^* \cdot \mathbf{x} + b^* = 0$

- 分类决策函数： $f(\mathbf{x}) = sign(\mathbf{w}^* \cdot \mathbf{x} + b^*) = sign\left(\sum_{i=1}^{n} \alpha_i^* y_i (\mathbf{x} \cdot \mathbf{x}_i) + b^*\right)$

软间隔支持向量：注意 $\alpha_j^* > 0$ 的样本点均称为支持向量！

图中，第一个点到其正确边界的距离为：$\dfrac{\xi_1}{\|\mathbf{w}\|}$，其它类推。

1： $\alpha^* = C$， $\xi_1 > 1$

2： $\alpha^* = C$， $\xi_2 > 1$

3： $\alpha^* = C$， $0 < \xi_3 < 1$

4： $\alpha^* = C$， $\xi_4 > 1$

5： $\alpha^* = C$， $0 < \xi_5 < 1$

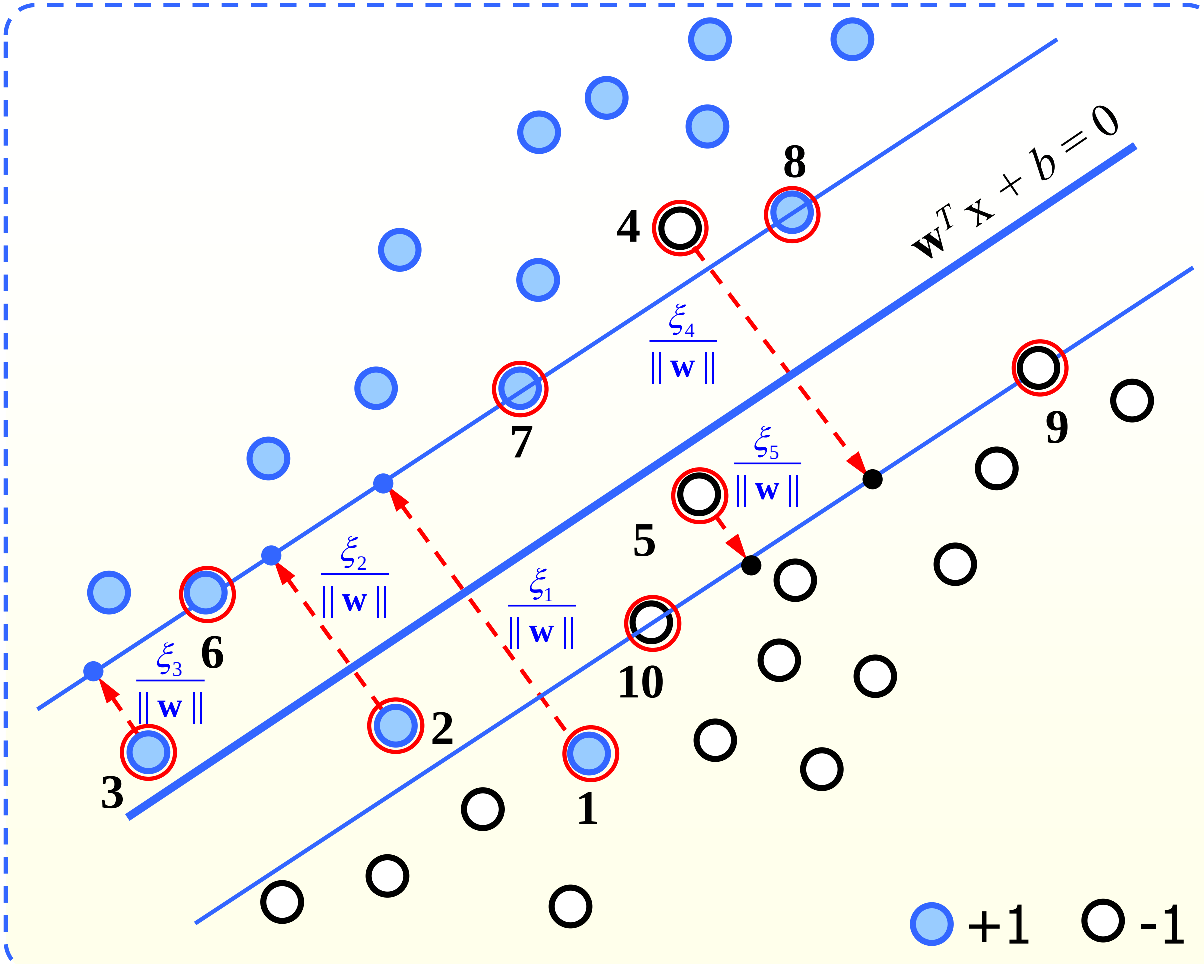
6： $0 < \alpha^* < C$， $\xi_6 = 0$

7： $0 < \alpha^* < C$， $\xi_7 = 0$

8： $0 < \alpha^* < C$， $\xi_8 = 0$

9： $0 < \alpha^* < C$， $\xi_9 = 0$

10： $0 < \alpha^* < C$， $\xi_{10} = 0$



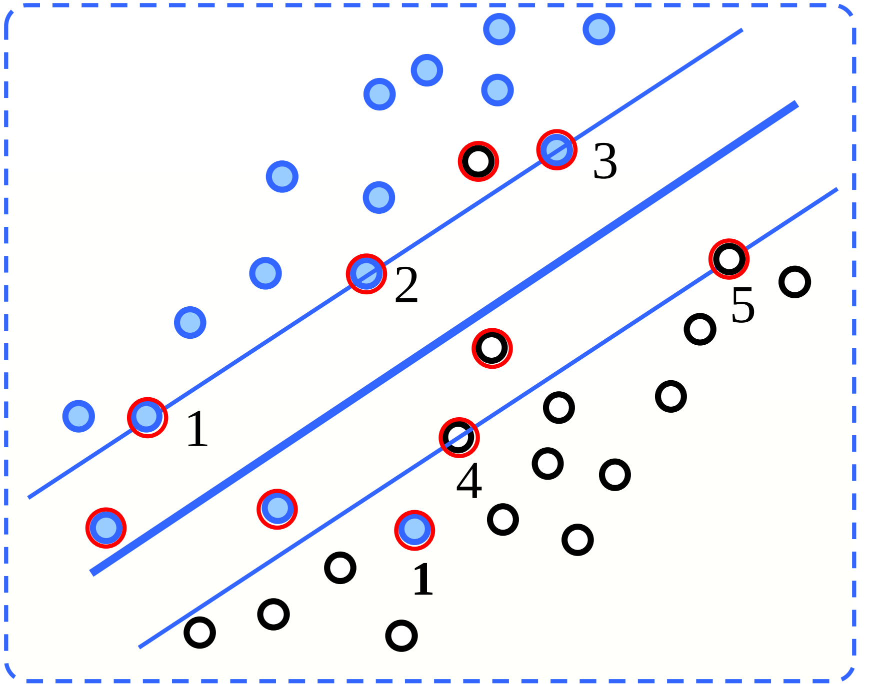
图中带红色圆圈的均表示支持向量

3.3 支持向量机

- **软间隔支持向量**
  - 支撑面以外（两个类边界以外）的样本点：$\alpha^* = 0$
  - 支持向量：$\alpha^* > 0$
    - 包含位于边界上的点，两个类边界以内的，以及错分点（边界以外）。
  - 位于类边界上的点其对应的拉格朗日乘子可能有如下三种情形：
    - $\alpha^* = 0$ (正好不是支持向量); $0 < \alpha^* < C$; or $\alpha^* = C$
- **模型求解：序列最小最优算法（略）**

3.3 支持向量机

- **偏置b的确定**
  - 不唯一



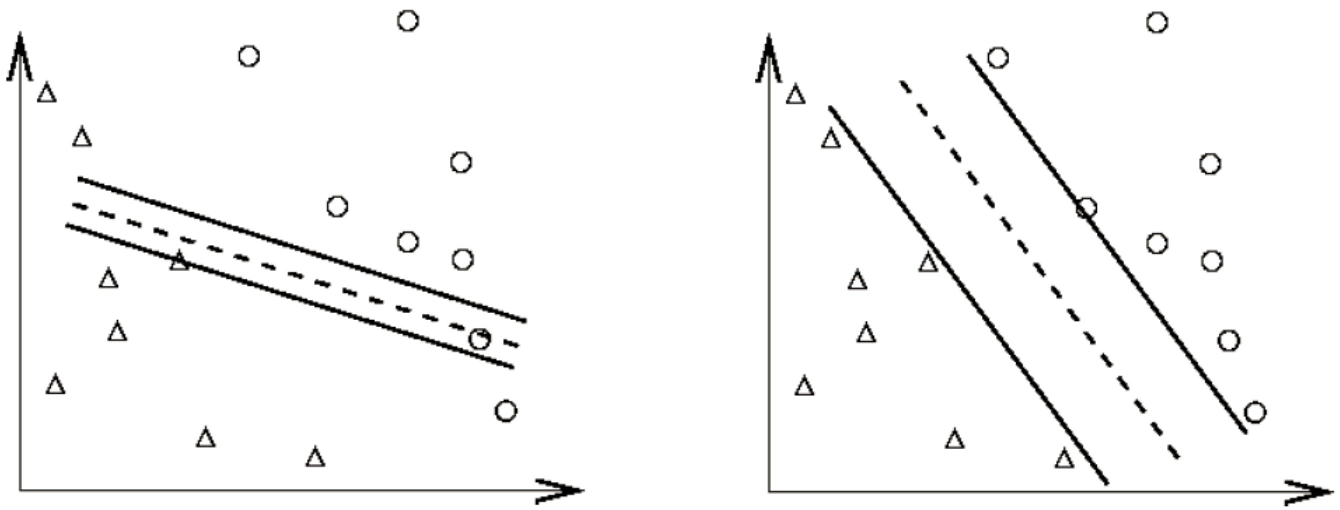
$$b^* = y_j - \sum_{i=1}^{n} y_i \alpha_i^* (\mathbf{x}_i \cdot \mathbf{x}_j), \quad 0 < \alpha_j^* < C$$

在所有符合条件的样本上计算一个b^*，然后取平均:

$$b^* = \frac{1}{\left|\{\alpha_k^* : 0 < \alpha_k^* < C\}\right|} \sum_{0<\alpha_k^*<C} (y_k - \sum_{i=1}^{n} \alpha_i^* y_i \mathbf{x}_i \cdot \mathbf{x}_j)$$

3.3 支持向量机

- **C的选择对分界面的影响**

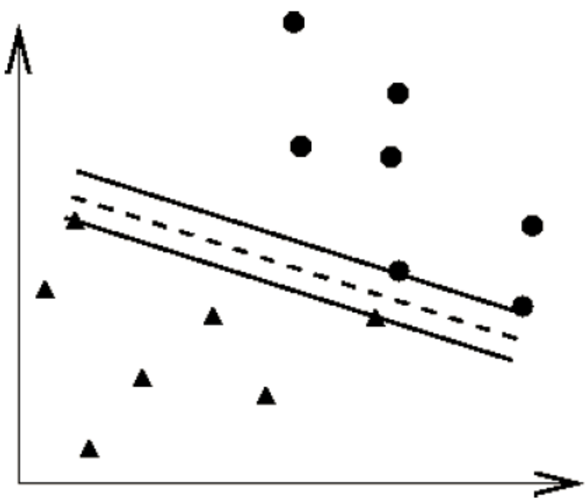


过拟合(overfitting)

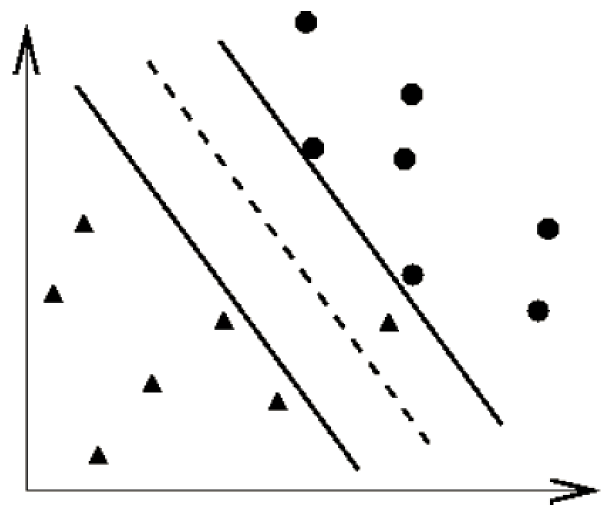
通过选择合适的C值，适当的注意分类间隔，能减少过拟合。

3.3 支持向量机

- **C的选择对分界面的影响**



C值较大，更加关心错分样本，倾向于产生没有错分样本的分界面



C值较小，更加关心分类间隔，倾向于产生大间隔的分界面

3.3 支持向量机

- **如何确定C？**

交叉验证（cross-validation）：将训练集分成p等份，依次进行p次分类器学习-分类器测试过程。

- 每次选择p-1份数据训练分类器（SVM模型），在剩下的1份数据集上进行测试。
- 交叉验证的正确率为p次测试的平均结果。

模型参数值设置的技术路线：通过对不同的C值进行交叉验证，取正确率最高的C值，在所有训练数据上重新学习SVM模型。

3.4 支持向量机回归

- **任务**
  - 给定n个训练样本$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)$，其中$\mathbf{x}_i \in \mathrm{R}^d$，$i$ =1,2,…,n为d维空间中的样本特征，$y_i \in R$为其对应的回归目标，希望学习到如下一个回归模型使得 $f(\mathbf{x})$ 与 y 尽可能地接近：

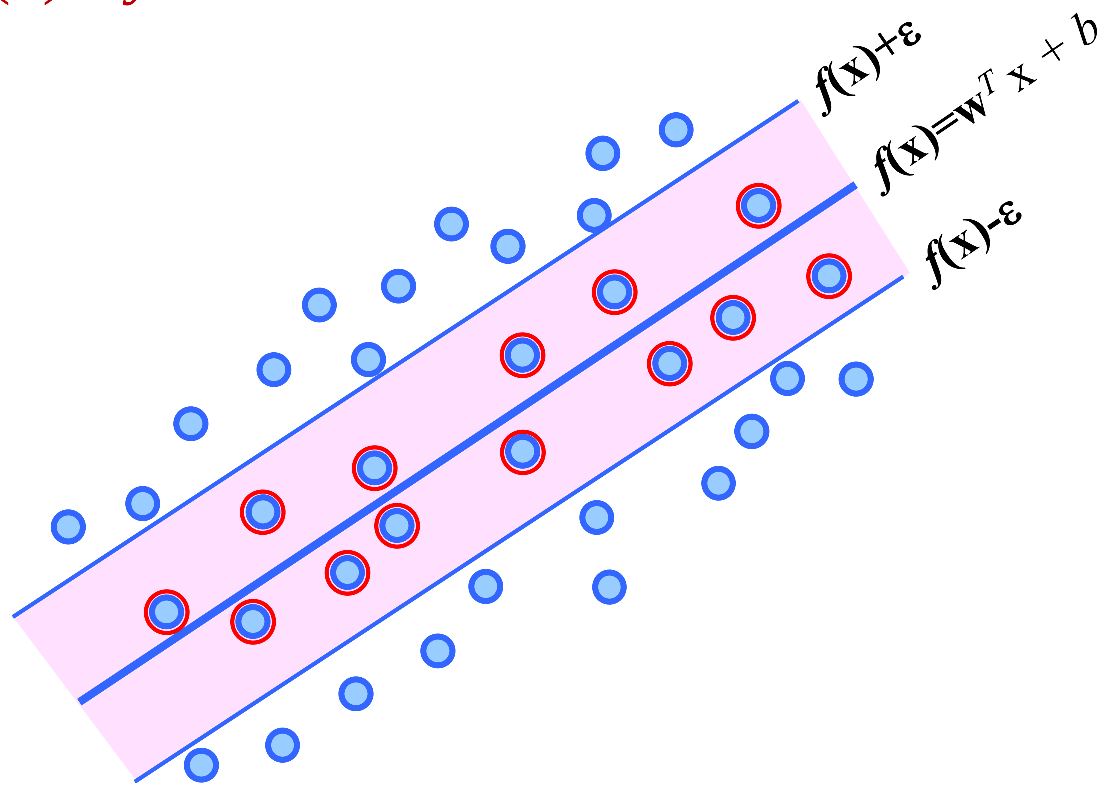
$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b, \quad \mathbf{x} \in R^d$$

  - 传统的线性最小二乘法 (正则化)：

$$\min_{\mathbf{w},\mathbf{b}} \sum_{i=1}^{n} \left(\mathbf{w}^T \mathbf{x}_i + b - y_i \right)^2 + \lambda \|\mathbf{w}\|_2^2$$

3.4 支持向量机回归

- Support vector regression, SVR
  - 假设$f(\mathbf{x})$与y之间可以有 ε 容忍偏差。



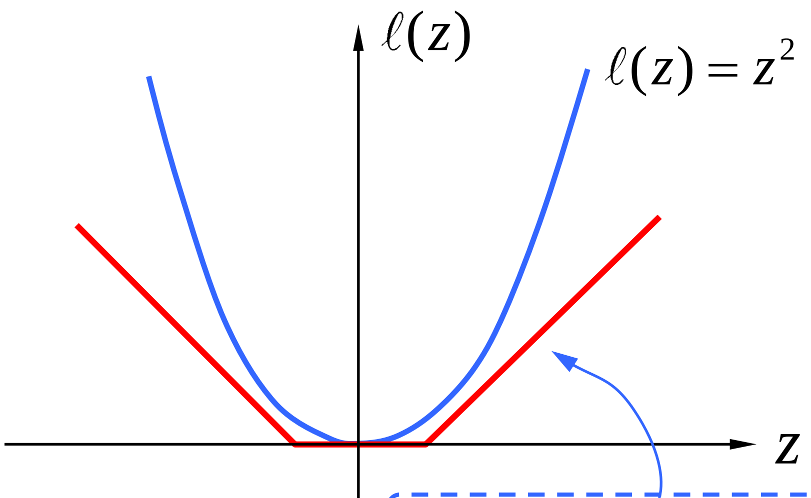
3.4 支持向量机回归

ε-insensitive loss function
ε-不敏感损失函数

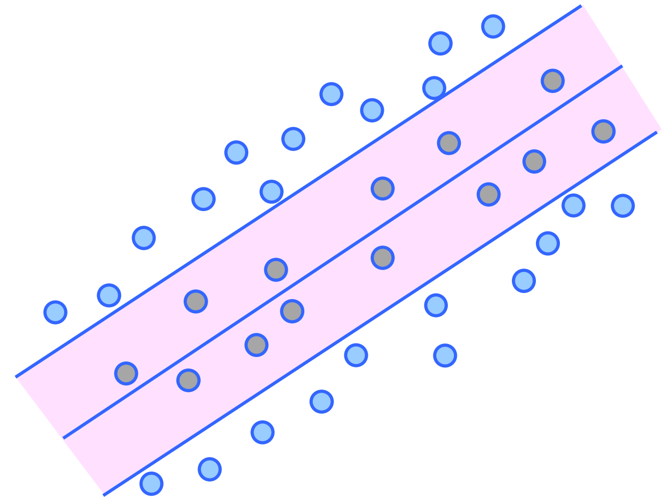
- **学习模型**

$$\min_{\mathbf{w},b} \quad \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\ell_\varepsilon\big(f(\mathbf{x}_i)-y_i\big),$$

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



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



3.4 支持向量机回归

-

$$s.t. \begin{cases} y_i - (w^T x_i + b) < \epsilon + \xi_i \\ (w^T x_i + b) - y_i < \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

$\leq \varepsilon$

wise

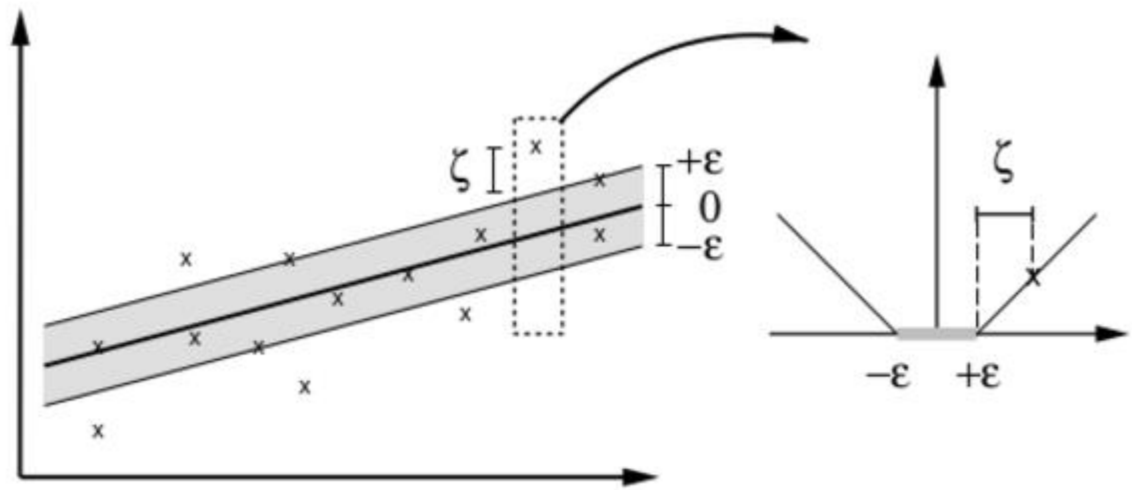


Figure 1 The soft margin loss setting corresponds for a linear SV machine.

$$\ell_{\varepsilon}(z) = \begin{cases} \\ |z| - \varepsilon, & \text{otherwise} \end{cases}$$

3.4 支持向量机回归

- **松弛模型**

$$
\begin{aligned}
\min_{\mathbf{w},b,\xi_i,\hat{\xi}_i} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\left(\xi_i + \hat{\xi}_i\right) \\
s.t. \quad & f(\mathbf{x}_i) - y_i \le \varepsilon + \hat{\xi}_i, \\
& y_i - f(\mathbf{x}_i) \le \varepsilon + \hat{\xi}_i, \\
& \xi_i \ge 0, \\
& \hat{\xi}_i \ge 0, \\
& i = 1, 2, \cdots, n
\end{aligned}
$$

3.4 支持向量机回归

- **松弛模型 的广义拉格朗日函数**

$$
\begin{aligned}
&L\left(\mathbf{w}, b, \boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}, \xi_i, \hat{\xi}_i, \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}\right) \\
&= \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\left(\xi_i + \hat{\xi}_i\right) - \sum_{i=1}^{n}\mu_i\xi_i - \sum_{i=1}^{n}\hat{\mu}_i\hat{\xi}_i \\
&\quad + \sum_{i=1}^{n}\alpha_i\left(f(\mathbf{x}_i) - y_i - \varepsilon - \xi_i\right) + \sum_{i=1}^{n}\hat{\alpha}_i\left(y_i - f(\mathbf{x}_i) - \varepsilon - \hat{\xi}_i\right)
\end{aligned}
$$

3.4 支持向量机回归

- **松弛模型求解**

$$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$$

$$\begin{aligned} & L\left(\mathbf{w}, b, \boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}, \xi_i, \hat{\xi}_i, \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}\right) \\ = & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\left(\xi_i + \hat{\xi}_i\right) - \sum_{i=1}^{n}\mu_i\xi_i - \sum_{i=1}^{n}\hat{\mu}_i\hat{\xi}_i \\ & + \sum_{i=1}^{n}\alpha_i\left(f(\mathbf{x}_i) - y_i - \varepsilon - \xi_i\right) + \sum_{i=1}^{n}\hat{\alpha}_i\left(y_i - f(\mathbf{x}_i) - \varepsilon - \hat{\xi}_i\right) \end{aligned}$$

令 $$\frac{\partial L\left(\mathbf{w}, b, \boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}, \xi_i, \hat{\xi}_i, \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}\right)}{\partial \mathbf{w}} = 0, \quad \frac{\partial L\left(\mathbf{w}, b, \boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}, \xi_i, \hat{\xi}_i, \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}\right)}{\partial b} = 0$$

3.4 支持向量机回归

- **松弛模型求解**

$$\frac{\partial L\left(\mathbf{w}, b, \boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}, \xi_i, \hat{\xi}_i, \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}\right)}{\partial \mathbf{w}} = 0$$

$$\frac{\partial L\left(\mathbf{w}, b, \boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}, \xi_i, \hat{\xi}_i, \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}\right)}{\partial b} = 0$$

$$\Rightarrow$$

$$\mathbf{w} = \sum_{i=1}^{n} \left(\alpha_i - \hat{\alpha}_i\right) \mathbf{x}_i$$

$$0 = \sum_{i=1}^{n} \left(\alpha_i - \hat{\alpha}_i\right)$$

$$C = \alpha_i + \mu_i$$

$$C = \hat{\alpha}_i + \hat{\mu}_i$$

3.4 支持向量机回归

- **松驰模型求解**

(代入)

$$L\left(\mathbf{w},b,\boldsymbol{\alpha},\hat{\boldsymbol{\alpha}},\xi_i,\hat{\xi}_i,\boldsymbol{\mu},\hat{\boldsymbol{\mu}}\right)$$

$$=\frac{1}{2}\|\mathbf{w}\|_2^2+C\sum_{i=1}^{n}\left(\xi_i+\hat{\xi}_i\right)-\sum_{i=1}^{n}\mu_i\xi_i-\sum_{i=1}^{n}\hat{\mu}_i\hat{\xi}_i$$

$$+\sum_{i=1}^{n}\alpha_i\left(f(\mathbf{x}_i)-y_i-\varepsilon-\xi_i\right)+\sum_{i=1}^{n}\hat{\alpha}_i\left(y_i-f(\mathbf{x}_i)-\varepsilon-\hat{\xi}_i\right)$$

$$\mathbf{w}=\sum_{i=1}^{n}\left(\alpha_i-\hat{\alpha}_i\right)\mathbf{x}_i$$

$$0=\sum_{i=1}^{n}\left(\alpha_i-\hat{\alpha}_i\right)$$

$$C=\alpha_i+\mu_i$$

$$C=\hat{\alpha}_i+\hat{\mu}_i$$

$$\max_{\boldsymbol{\alpha},\hat{\boldsymbol{\alpha}}}\ \sum_{i=1}^{n}\left(y_i\left(\hat{\alpha}_i-\alpha_i\right)-\varepsilon\left(\hat{\alpha}_i+\alpha_i\right)\right)$$

$$-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\hat{\alpha}_i-\alpha_i\right)\left(\hat{\alpha}_j-\alpha_j\right)\mathbf{x}_i^T\mathbf{x}_j$$

$$s.t.\quad \sum_{i=1}^{n}\left(\hat{\alpha}_i-\alpha_i\right)=0,$$

$$0\le\hat{\alpha}_i,\alpha_i\le C.$$

3.4 支持向量机回归

- **回顾KKT条件**

原最优化问题：

$$
\begin{aligned}
\min \quad & f(\mathbf{x}) \\
s.t. \quad & h_j(\mathbf{x}) = 0, \quad j = 1, 2, \cdots, p \\
& g_k(\mathbf{x}) \le 0, \quad k = 1, 2, \cdots, q
\end{aligned}
$$

$\Rightarrow$

KKT条件

$$
\begin{aligned}
& \frac{\partial L(\mathbf{x}, \lambda, \mu)}{\partial \mathbf{x}} = \mathbf{0}, \\
& \lambda_j \ne 0, \\
& \mu_k \ge 0, \\
& \mu_k g_k(\mathbf{x}) = 0, \\
& h_j(\mathbf{x}) = 0, \\
& g_k(\mathbf{x}) \le 0, \\
& \quad j = 1, 2, \cdots, p \\
& \quad k = 1, 2, \cdots, q
\end{aligned}
$$

拉格朗日方程：

$$
L(\mathbf{x}, \lambda, \mu) = f(\mathbf{x}) + \sum_{j=1}^{p} \lambda_j h_j(\mathbf{x}) + \sum_{k=1}^{q} \mu_k g_k(\mathbf{x})
$$

3.4 支持向量机回归

- **松弛模型求解**

（满足右则KKT条件）

$$
\begin{aligned}
&L\left(\mathbf{w}, b, \boldsymbol{\alpha}, \hat{\boldsymbol{\alpha}}, \xi_i, \hat{\xi}_i, \boldsymbol{\mu}, \hat{\boldsymbol{\mu}}\right) \\
&= \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\left(\xi_i + \hat{\xi}_i\right) - \sum_{i=1}^{n}\mu_i\xi_i - \sum_{i=1}^{n}\hat{\mu}_i\hat{\xi}_i \\
&\quad + \sum_{i=1}^{n}\alpha_i\left(f(\mathbf{x}_i) - y_i - \varepsilon - \xi_i\right) \\
&\quad + \sum_{i=1}^{n}\hat{\alpha}_i\left(y_i - f(\mathbf{x}_i) - \varepsilon - \hat{\xi}_i\right)
\end{aligned}
$$

$$
\Rightarrow
\begin{cases}
\alpha_i\left(f(\mathbf{x}_i) - y_i - \varepsilon - \xi_i\right) = 0, \\
\hat{\alpha}_i\left(y_i - f(\mathbf{x}_i) - \varepsilon - \hat{\xi}_i\right) = 0, \\
\alpha_i\hat{\alpha}_i = 0, \\
\xi_i\hat{\xi}_i = 0, \\
(C - \alpha_i)\xi_i = 0, \\
(C - \hat{\alpha}_i)\hat{\xi}_i = 0
\end{cases}
$$

3.4 支持向量机回归

- **松弛模型求解**

$$\begin{cases}\alpha_i\left(f(\mathbf{x}_i)-y_i-\varepsilon-\xi_i\right)=0,\\ \hat{\alpha}_i\left(y_i-f(\mathbf{x}_i)-\varepsilon-\hat{\xi}_i\right)=0,\\ \alpha_i\hat{\alpha}_i=0,\\ \xi_i\hat{\xi}_i=0,\\ (C-\alpha_i)\xi_i=0,\\ (C-\hat{\alpha}_i)\hat{\xi}_i=0\end{cases}$$

$$f(\mathbf{x}_i)-y_i-\varepsilon-\xi_i=0 \quad \Leftrightarrow \quad \alpha_i>0$$

$$y_i-f(\mathbf{x}_i)-\varepsilon-\hat{\xi}_i=0 \quad \Leftrightarrow \quad \hat{\alpha}_i>0$$

✓ 当且仅当 (x_i, y_i)不落入ε间隔带时，相应的α_i和$\hat{\alpha}_i$取非零值。

✓ 约束条件：$\begin{cases}f(\mathbf{x}_i)-y_i-\varepsilon-\xi_i=0\\ y_i-f(\mathbf{x}_i)-\varepsilon-\hat{\xi}_i=0\end{cases}$ 不能同时成立，所以 α_i 和 $\hat{\alpha}_i$ 至少有一个为零。

3.4 支持向量机回归

- **最后的解：**

$$\mathbf{w}=\sum_{i=1}^{n}(\alpha_i-\hat{\alpha}_i)\mathbf{x}_i \quad \Rightarrow \quad f(\mathbf{x})=\mathbf{w}^T\mathbf{x}+b=\sum_{i=1}^{n}(\hat{\alpha}_i-\alpha_i)\mathbf{x}^T\mathbf{x}_i+b$$

$$\begin{cases}
\boxed{\alpha_i\left(f(\mathbf{x}_i)-y_i-\varepsilon-\xi_i\right)=0,} \\
\hat{\alpha}_i\left(y_i-f(\mathbf{x}_i)-\varepsilon-\hat{\xi}_i\right)=0, \\
\alpha_i\hat{\alpha}_i=0, \\
\xi_i\hat{\xi}_i=0, \\
\boxed{(C-\alpha_i)\xi_i=0,} \\
(C-\hat{\alpha}_i)\hat{\xi}_i=0
\end{cases}$$

✓ 在获得α_i后，如果0< α_i <C，则必有ξ_i =0，从而：

$$b=y_i+\varepsilon-\sum_{i=1}^{n}(\hat{\alpha}_i-\alpha_i)\mathbf{x}^T\mathbf{x}_i$$

注：b可以取多个点的平均

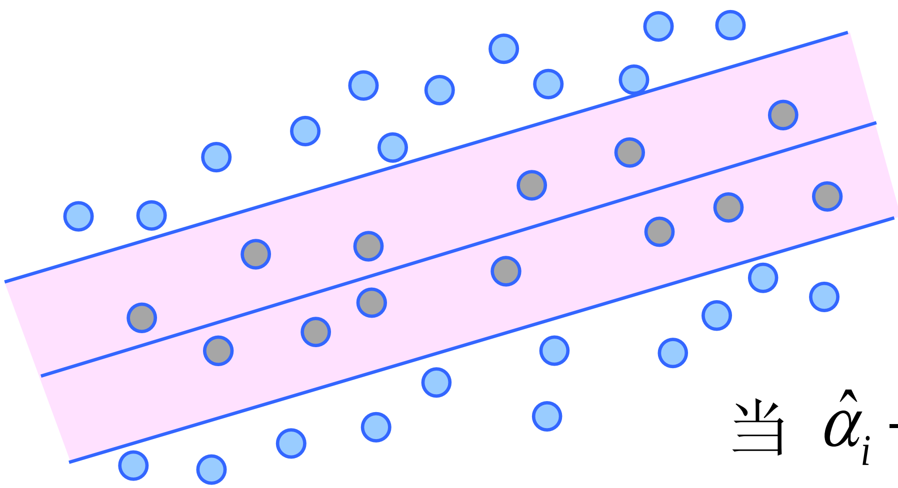
3.4 支持向量机回归

- **支持向量**

$$f(\mathbf{x}_i) - y_i - \varepsilon - \xi_i = 0 \quad \Leftrightarrow \quad \alpha_i > 0$$

$$y_i - f(\mathbf{x}_i) - \varepsilon - \hat{\xi}_i = 0 \quad \Leftrightarrow \quad \hat{\alpha}_i > 0$$

上述条件有一个成立即表示该点必定落在ε间隔带之外。



当 $\hat{\alpha}_i - \alpha_i \neq 0$ 时，所对应的点为支持向量，它们必定落在ε间隔带之外。

3.5 支持向量机排序

- **Learning to rank**
  - 排序一直是信息检索的核心问题之一，Learning to Rank(简称LTR)用机器学习的思想来解决排序问题。
  - L2R 有三种主要的方法：PointWise，PairWise，ListWise。
  - Ranking SVM 算法是PointWise方法的一种，由R. Herbrich等人在2000提出。
  - RankSVM的基本思想是，将排序问题转化为pairwise的分类问题，然后使用SVM分类模型进行学习并求解。

3.5 支持向量机排序

- **将排序问题转化为分类问题**
  - 不失一般性，以文档查询为背景“query-doc pair”。
  - 记一个文档的特征为$\mathbf{x}$，我们的目的是需要找到一个排序函数$f(\mathbf{x})$，根据$f(\mathbf{x})$的大小来决定排序顺序。即如果$f(\mathbf{x}_i)>f(\mathbf{x}_j)$，则 $\mathbf{x}_i$ 应该排在 $\mathbf{x}_j$ 的前面，反之亦然：

$$\mathbf{x}_i \succ \mathbf{x}_j \quad \Leftrightarrow \quad f(\mathbf{x}_i) > f(\mathbf{x}_j)$$

  - 理论上，$f(\mathbf{x})$可以是任意函数。
  - 为了简单起见，假设其为线性函数：$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$
  - 由于排序不受参数b的影响，所以可令：$f(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$

3.5 支持向量机排序

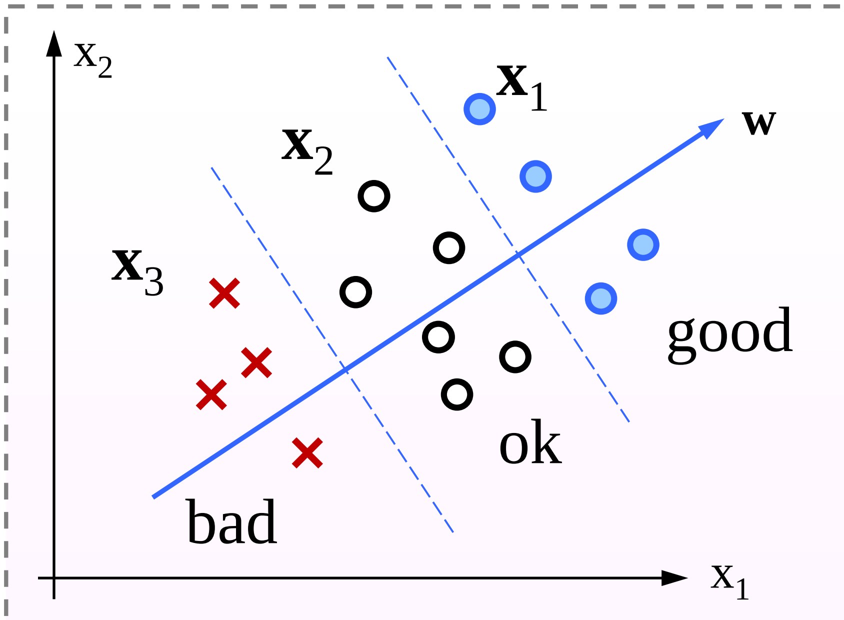
- **为什么可以转换为两类分类问题？**
  - 首先，对于任意两个数据点 $\mathbf{x}_i$ 和 $\mathbf{x}_j$，若 $f(\mathbf{x})$是线性函数，则如下关系成立：

$$f(\mathbf{x}_i) > f(\mathbf{x}_j) \quad \Leftrightarrow \quad \mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_j) > 0$$

  - 然后，可以对$\mathbf{x}_i$和$\mathbf{x}_j$的差值向量引入两类分类问题，按如下方式进行标签赋值：

$$y = \begin{cases} +1, & \text{if } x_i \succ x_j \\ -1, & \text{if } x_i \prec x_j \end{cases}, \qquad \mathbf{w}^T(\mathbf{x}_i - \mathbf{x}_j) > 0 \quad \Leftrightarrow \quad y = +1$$

- **SVM模型解决排序问题**
  - 将排序问题转化为分类问题之后，可使用Linear SVM或 kernel SVM解决排序问题。



上图展示了一组查询，给出了所召回的文档，其中文档的相关程度等级分为三档(good, ok, bad)。权重向量**w**对应排序函数，可以对“查询-返回”对进行打分和排序。

- **SVM模型解决排序问题**
  - 为SVM准备样本：

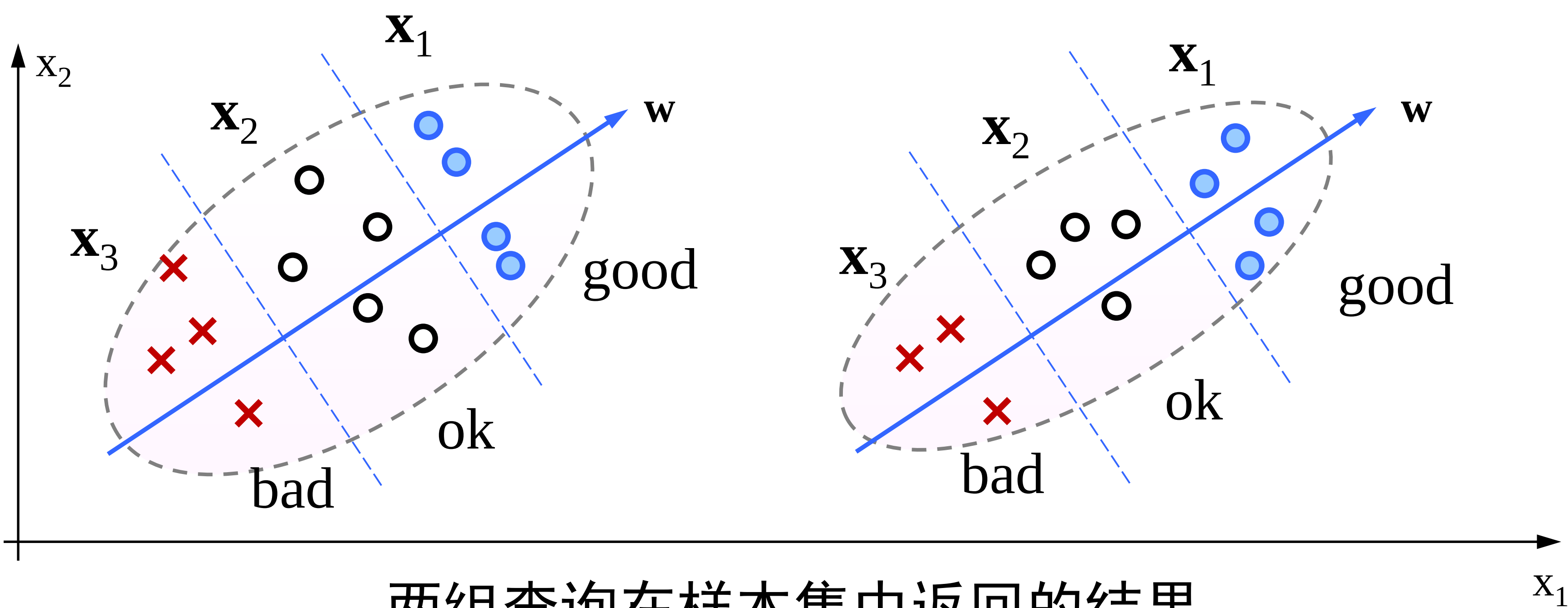
✓ 给定一个查询及其反馈，可对样本进行组合，形成新数据点：$\mathbf{x}_1-\mathbf{x}_2$，$\mathbf{x}_1-\mathbf{x}_3$，$\mathbf{x}_2-\mathbf{x}_3$。其label也会被重新赋值，比如将$\mathbf{x}_1-\mathbf{x}_2$，$\mathbf{x}_1-\mathbf{x}_3$，$\mathbf{x}_2-\mathbf{x}_3$的label赋值为正类。

✓ 为了构造分类问题，还需负样本。可以使用其反方向向量作为负样本：$\mathbf{x}_2-\mathbf{x}_1$，$\mathbf{x}_3-\mathbf{x}_1$，$\mathbf{x}_3-\mathbf{x}_2$。

✓ 需要指出的是，在组合形成新样本时，不能使用在原始排序问题中处于相同相似度等级的两个数据点，也不能使用处于不同query下的两个数据点来组合新样本。

SVM模型解决排序问题

- 为SVM准备样本：

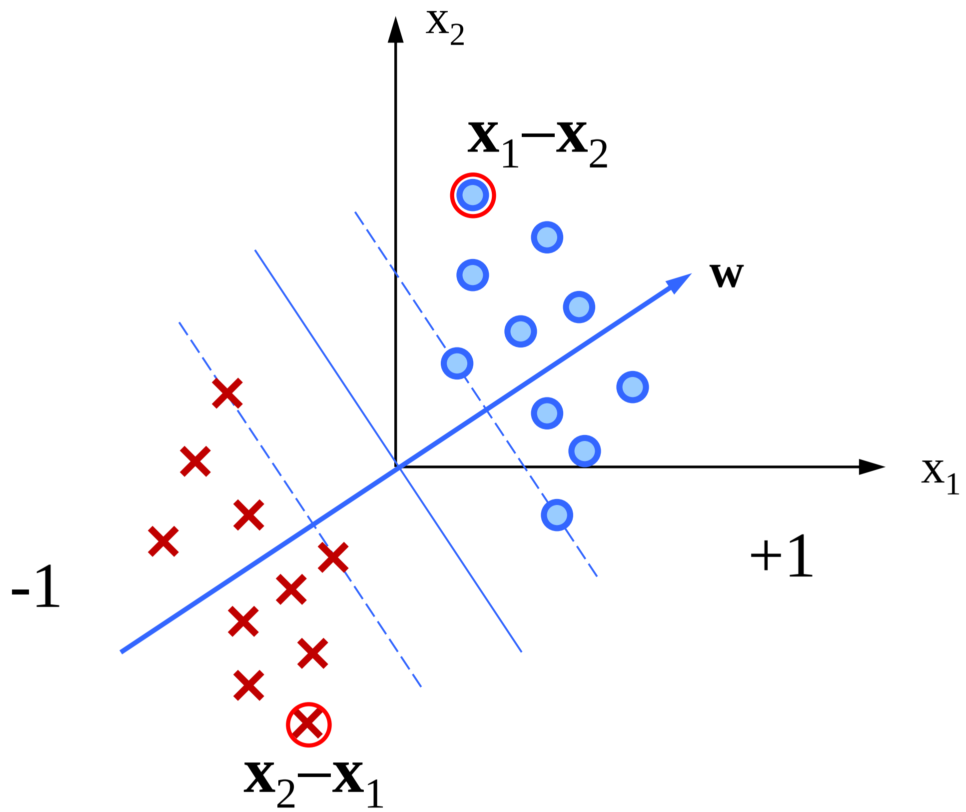


两组查询在样本集中返回的结果

正样本： $\mathbf{x}_1-\mathbf{x}_2$， $\mathbf{x}_1-\mathbf{x}_3$， $\mathbf{x}_2-\mathbf{x}_3$

负样本： $\mathbf{x}_2-\mathbf{x}_1$， $\mathbf{x}_3-\mathbf{x}_1$， $\mathbf{x}_3-\mathbf{x}_2$

- **SVM模型解决排序问题**
  - 为SVM准备样本：



3.5 支持向量机排序

- **学习模型**
  - 转化为分类问题后，便可以采用SVM的通用方式进行求解。学习模型如下：

$$\begin{aligned} \min_{\mathbf{w},b,\xi} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ s.t. \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \\ & i = 1, 2, \cdots, n \end{aligned}$$

支持向量机分类

$$\begin{aligned} \min_{\mathbf{w},\xi} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\ s.t. \quad & y_i\left(\mathbf{w}^T\left(\mathbf{x}_i^{(1)} - \mathbf{x}_i^{(2)}\right)\right) \geq 1 - \xi_i, \\ & \xi_i \geq 0, \\ & i = 1, 2, \cdots, n \end{aligned}$$

支持向量机排序

3.5 支持向量机排序

- **学习模型**
  - 类似地采用合页损失函数：

$$\min_{\mathbf{w},b} \sum_{i=1}^{n} \left[1 - y_i \left(\mathbf{w}^T \left(\mathbf{x}_i^{(1)} - \mathbf{x}_i^{(2)} \right) \right) \right]_+ + \lambda \| \mathbf{w} \|_2^2$$

$$\text{where } [z]_+ = \begin{cases} z, & \text{if } z > 0 \\ 0, & \text{otherwise} \end{cases}$$

3.5 支持向量机排序

- **对偶学习模型**

支持向量机分类

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i \cdot \mathbf{x}_j) - \sum_{i=1}^{n}\alpha_i$$

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

$$0 \le \alpha_i \le C, \quad i = 1, 2, \ldots, n$$

$$\Downarrow$$

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j\left(\mathbf{x}_i^{(1)} - \mathbf{x}_i^{(2)}\right)\cdot\left(\mathbf{x}_j^{(1)} - \mathbf{x}_j^{(2)}\right) - \sum_{i=1}^{n}\alpha_i$$

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

$$0 \le \alpha_i \le C, \quad i = 1, 2, \ldots, n$$

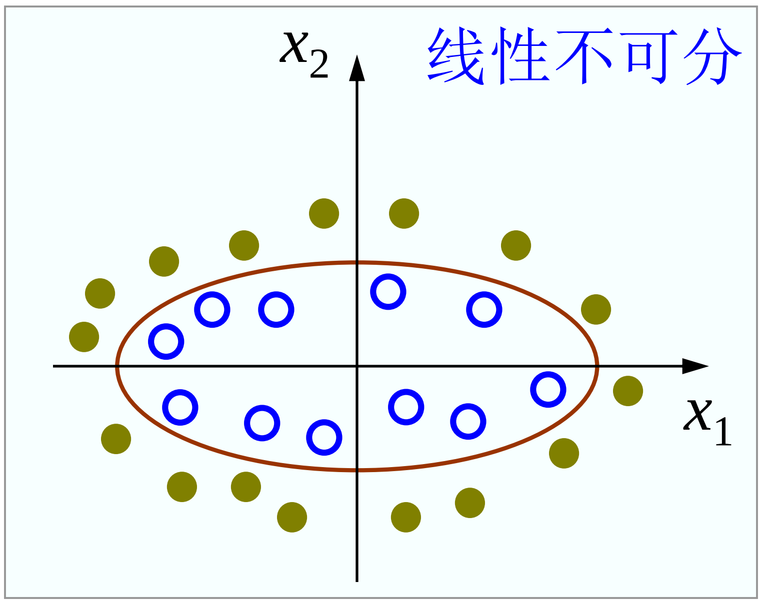
支持向量机排序

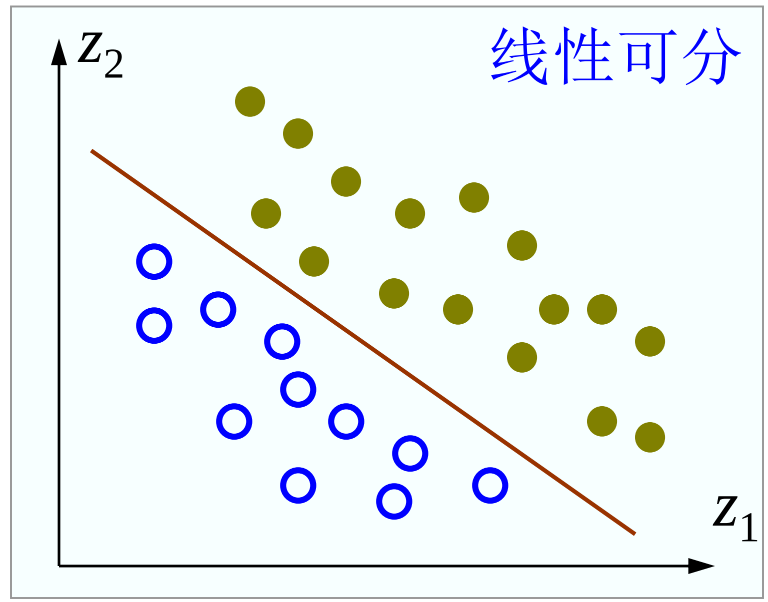
3.5 支持向量机排序

- **其它改进**
  - Hinge Loss是0-1损失。损失函数的优化目标可以进一步与Information Retrieval的Evaluation常用指标建立紧密联系。
  - **更复杂的方法：采用**Ordinal Regression方法来对此问题进行建模。

3.6 核技巧

- **非线性分类问题**





椭圆：$w_1 x_1^2 + w_2 x_2^2 + b = 0$ ⇨ 直线：$w_1 z_1 + w_2 z_2 + b = 0$

变换： $\mathbf{z}=\phi(\mathbf{x})=((x_1)^2, (x_2)^2)^T$

3.6 核技巧

- **用线性方法解决非线性问题**
  - **第一步，**使用一个变换**将原空间中的数据映射到新空间**
  - **第二步，在新空间里用线性分类学习方法**从训练中学习一个分类模型

核技巧就是属于这样的方法！

3.6 核技巧

- **表示定理**
  - 令H为核函数K对应的再生核希尔伯特空间，$\|h\|_H$表示H空间中关于h的范数，对任意单调递增函数$\Omega:[0,\infty]\to R$和任意非负损失函数loss：$R^m\to[0,\infty]$，优化问题

$$\min_{h\in H}\ F(h)=\Omega\left(\|h\|_H\right)+loss\left(h(\mathbf{x}_1),h(\mathbf{x}_2),\cdots,h(\mathbf{x}_m)\right)$$

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

表示定理对损失函数没有限制，对正则化项Ω仅要求单调递增，甚至不要求Ω是凸函数。因此，对于一般的损失函数和正则项，优化问题的解都可以表示为核函数$K(\mathbf{x},\mathbf{x}_i)$的线性组合。（威力）

3.6 核技巧

- **正定核**
  - **已知映射**ϕ (**x**),可以通过求ϕ (**x**)和ϕ (**y**)的内积得到核函数 K (**x**, **y**)
  - **不用构造映射ϕ (x)，能否直接判断一个给定的 K (x, y) 是不是核函数？**
  - K (**x**, **y**) 满足什么条件才能成为核函数？
  - **假定**K (**x**, **y**) 是$X \times X$上的对称函数，并且对于任意的$\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_m \in$ X, K (**x**, **y**) 关于$\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_m$的Gram矩阵是半正定的，则可以依据K (**x**, **y**) 构造一个希尔伯特空间。

3.6 核技巧

- **正定核的充要条件**
  - 设 $K: X \times X \rightarrow$R对称函数(定义在$X \times X$ 上)，则$K(\mathbf{x}, \mathbf{y})$为正定核的充要条件是对任意 $\mathbf{x}_i \in X$, $i=1,2,\ldots,m$, $K(\mathbf{x}, \mathbf{y})$对应的Gram矩阵：

$$K = \left[K(\mathbf{x}_i, \mathbf{x}_j) \right] \in R^{m \times m},$$

是半正定矩阵。

Property: Any **symmetric positive definite** matrix specifies a kernel matrix & every kernel matrix is **symmetric positive definite**

3.6 核技巧

- **Mercer核**
  - 设 $K: X \times X \to \mathrm{R}$ 是对称函数，$K(\mathbf{x},\mathbf{y})$为某个特征空间的内积运算的充要条件是，对任意的非零函数 $\phi(\mathbf{x})$, 且$\phi(\mathbf{x})$平方可积，有

$$\iint K(\mathbf{x},\mathbf{y})\phi(\mathbf{x})\phi(\mathbf{y})d\mathbf{x}d\mathbf{y} > 0.$$

此时，$K(\mathbf{x},\mathbf{y})$为Mercer核。

正定核比Mercer核更具一般性！

3.6 核技巧

- **常用核函数**

线性核： $$K_{lin}(\mathbf{x},\mathbf{y}) = \mathbf{x}\cdot\mathbf{y}$$

多项式核： $$K_{pol}(\mathbf{x},\mathbf{y}) = (\mathbf{x}\cdot\mathbf{y}+1)^p$$

径向基函数核： $$K_{Gau}(\mathbf{x},\mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right)$$

3.7 KSVM

- 从对偶问题直接实现SVM核化—训练

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j(\mathbf{x}_i\cdot\mathbf{x}_j)-\sum_{i=1}^{n}\alpha_i$$

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

$$\alpha_i \geq 0, \quad i=1,2,\ldots,n$$

$$\Downarrow$$

$$\min_{\boldsymbol{\alpha}} \quad \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j K(\mathbf{x}_i,\mathbf{x}_j)-\sum_{i=1}^{n}\alpha_i$$

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

$$\alpha_i \geq 0, \quad i=1,2,\ldots,n$$

3.7 KSVM

- **预测（对新数据）**

$$f(\mathbf{x}) = sign\left(\sum_{i=1}^{n} \alpha_i^* y_i (\mathbf{x} \cdot \mathbf{x}_i) + b^*\right), \quad b^* = y_j - \sum_{i=1}^{n} \alpha_i^* y_i (\mathbf{x}_i \cdot \mathbf{x}_j)$$

$$\Downarrow$$

$$f(\mathbf{x}) = sign\left(\sum_{i=1}^{n} \alpha_i^* y_i K(\mathbf{x} \cdot \mathbf{x}_i) + b^*\right), \quad b^* = y_j - \sum_{i=1}^{n} \alpha_i^* y_i K(\mathbf{x}_i \cdot \mathbf{x}_j)$$

3.8 KPCA (此后几节不讲，感兴趣的同学自学)

- **主要文献**
  - B. Scholkopf, A. J. Smola, K. R. Muller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. Neural Computation,10(5):1299–1319, 1998.

3.8 KPCA

- **PCA**
  - 给定 $\mathbf{X}=[\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_n]\in \mathrm{R}^{n\times m}$，**并假定均值为零**
  - 计算协方差矩阵：$\mathbf{C}=\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^T$
  - 对 C 施行矩阵特征值分解：$\mathbf{C}=\mathbf{U\Sigma U}^T$
  - 取**指定个数的**最大特征值对应的特征向量子集，组成投影向量 $\mathbf{W}=\mathbf{U}s$,（$\mathbf{U}s$为$\mathbf{U}$的子矩阵）
  - 对新样本 $\mathbf{x}$, 将其投影至低维子空间：$\mathbf{y}=\mathbf{W}^T\mathbf{x}$

3.8 KPCA

- **KPCA**
  - 引入非线性映射： $\phi : \mathrm{R}^m \rightarrow F$
  - 将数据进行映射： $\{\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \ldots, \phi(\mathbf{x}_n)\} \subset F$
  - 计算协方差矩阵： $\bar{\mathbf{C}} = \frac{1}{n}\sum_{i=1}^{n}\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T$
  - 作特征值分解： $\bar{\mathbf{C}} = \bar{\mathbf{U}}\,\bar{\boldsymbol{\Sigma}}\,\bar{\mathbf{U}}^T$
  - **What are the difficulties?**

3.8 KPCA

- **考虑特征值分解问题**

一个标量（即数据在该特征向量上的投影）

$$\boxed{\lambda \mathbf{v} = \mathbf{C}\mathbf{v}}$$

且： $\mathbf{C}\mathbf{v} = \left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i\mathbf{x}_i^T\right)\mathbf{v} = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i^T\mathbf{v})\mathbf{x}_i = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}_i\cdot\mathbf{v})\mathbf{x}_i$

每个特征向量 $\mathbf{v}$ 均位于由 n 个数据点张成的子空间内！

$$\lambda \mathbf{v} = \mathbf{C}\mathbf{v} \Rightarrow \lambda(\mathbf{x}_i\cdot\mathbf{v}) = (\mathbf{x}_i\cdot\mathbf{C}\mathbf{v}),\quad i = 1, 2, \cdots, n$$

3.8 KPCA

- **考虑特征空间**

$$\boxed{\lambda \mathbf{V} = \bar{\mathbf{C}}\mathbf{V}}$$

$$\lambda \mathbf{V} = \bar{\mathbf{C}}\mathbf{V} \leftrightarrow \lambda(\phi(\mathbf{x}_i)\cdot \mathbf{V}) = (\phi(\mathbf{x}_i)\cdot \bar{\mathbf{C}}\mathbf{V}), \quad i = 1, 2, \cdots, n$$

每个特征向量 v 均位于由 n 个数据点{ $\phi(\mathbf{x}_1)$, $\phi(\mathbf{x}_2)$,…, $\phi(\mathbf{x}_n)$ }张成的子空间内.

因此，存在 α_i, $i = 1, 2, \ldots, n$, 使得 $\mathbf{v} = \sum_{i=1}^{n} \alpha_i \phi(\mathbf{x}_i)$.

同时，给定一组不同的系数，可以得到不同的v!

需要求解α_i！

3.8 KPCA

$$\mathbf{v}=\sum_{k=1}^{n}\alpha_k\phi(\mathbf{x}_k)$$

于是有

$$\lambda(\phi(\mathbf{x}_i)\cdot\mathbf{v})=(\phi(\mathbf{x}_i)\cdot\bar{\mathbf{C}}\cdot\mathbf{v}),\quad i=1,2,\cdots,n$$

$$\lambda\sum_{k=1}^{n}\alpha_k(\phi(\mathbf{x}_i)\cdot\phi(\mathbf{x}_k))$$

$$=\frac{1}{n}\sum_{k=1}^{n}\alpha_k\phi(\mathbf{x}_i)\cdot\sum_{j=1}^{n}\phi(\mathbf{x}_j)(\phi(\mathbf{x}_j)\cdot\phi(\mathbf{x}_k)),$$

$$i=1,2,\cdots,n$$

$$\lambda\sum_{k=1}^{n}\alpha_k(\phi(\mathbf{x}_i)\cdot\phi(\mathbf{x}_k))$$

$$=\left(\phi(\mathbf{x}_i)\cdot\frac{1}{n}\left(\sum_{j=1}^{n}(\phi(\mathbf{x}_j)\phi(\mathbf{x}_j)^T)\right)\sum_{k=1}^{n}\alpha_k\phi(\mathbf{x}_k)\right)$$

$$=\frac{1}{n}\sum_{j=1}^{n}(\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)\phi(\mathbf{x}_j)^T)\sum_{k=1}^{n}\alpha_k\phi(\mathbf{x}_k)$$

$$=\frac{1}{n}\sum_{j=1}^{n}(\phi(\mathbf{x}_j)^T[\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)])\sum_{k=1}^{n}\alpha_k\phi(\mathbf{x}_k)$$

$$=\frac{1}{n}\sum_{j=1}^{n}(\phi(\mathbf{x}_j)^T\sum_{k=1}^{n}\alpha_k[\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)]\phi(\mathbf{x}_k))$$

$$=\frac{1}{n}\sum_{j=1}^{n}(\phi(\mathbf{x}_j)^T\sum_{k=1}^{n}\alpha_k[\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)]\phi(\mathbf{x}_k))$$

$$=\frac{1}{n}\sum_{k=1}^{n}\sum_{j=1}^{n}\alpha_k[\phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)](\phi(\mathbf{x}_j)^T\phi(\mathbf{x}_k))$$

$$=\frac{1}{n}\sum_{k=1}^{n}\alpha_k\phi(\mathbf{x}_i)\cdot\sum_{j=1}^{n}\phi(\mathbf{x}_j)(\phi(\mathbf{x}_j)\cdot\phi(\mathbf{x}_k))$$

组合

定义 $n\times n$ 矩阵$\mathbf{K}$, $K_{ij}=\phi(\mathbf{x}_1)\cdot\phi(\mathbf{x}_2)$

$n\lambda\mathbf{K}\boldsymbol{\alpha}=\mathbf{K}\mathbf{K}\boldsymbol{\alpha}$, 即 $n\lambda\boldsymbol{\alpha}=\mathbf{K}\boldsymbol{\alpha}$

3.8 KPCA

- **对新样本x**
  - 将 x 映射至特征空间 F：$\mathbf{x} \to \phi(\mathbf{x})$
  - 取出指定维数子空间，即 F 中 eignevectors张成的子空间
  - 将 $\phi(\mathbf{x})$ 向该子空间进行投影，**比如投影后的第 k 个分量（即作非线性变换之后向第k 个特征向量投影）：**

$$\left(\mathbf{v}_k \cdot \phi(\mathbf{x})\right) = \sum_{i=1}^{n} \alpha_i^k \left(\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x})\right) = \sum_{i=1}^{n} \alpha_i^k K(\mathbf{x}_i, \mathbf{x})$$

$$\left(k = 1, 2, \cdots, d\right)$$

- **对新样本x**
  - Totally, 向d个方向进行投影，全部出来：

向第k个投影，得第k个系数：

$$\left(\mathbf{v}_k^T \cdot \phi(\mathbf{x})\right) = \sum_{i=1}^{n} \alpha_i^k \left(\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x})\right) = \sum_{i=1}^{n} \alpha_i^k K(\mathbf{x}_i, \mathbf{x})$$

Totally:

$$\mathbf{x} \rightarrow \mathbf{y} = \left[\mathbf{v}_1^T \phi(\mathbf{x}), \quad \mathbf{v}_2^T \phi(\mathbf{x}), \quad \cdots, \quad \mathbf{v}_d^T \phi(\mathbf{x})\right]^T$$

$\Downarrow$ 完全写出来

$$\mathbf{x} \in R^m \rightarrow \mathbf{y} = \left[\underbrace{\sum_{i=1}^{n} \alpha_i^1 K(\mathbf{x}_i, \mathbf{x})}_{\text{向第1个投影}}, \sum_{i=1}^{n} \alpha_i^2 K(\mathbf{x}_i, \mathbf{x}), \cdots, \underbrace{\sum_{i=1}^{n} \alpha_i^d K(\mathbf{x}_i, \mathbf{x})}_{\text{向第}d\text{个投影}}\right]^T \in R^d$$

对新样本x (再解释)：

$$\mathbf{x} \rightarrow \mathbf{y} = \left[\mathbf{v}_1^T\phi(\mathbf{x}), \quad \mathbf{v}_2^T\phi(\mathbf{x}), \quad \cdots, \quad \mathbf{v}_d^T\phi(\mathbf{x})\right]^T$$

$$\mathbf{x} \in R^m \rightarrow \left[\underbrace{\sum_{i=1}^{n}\alpha_i^1 K(\mathbf{x}_i, \mathbf{x})}_{\text{向第1个投影}}, \sum_{i=1}^{n}\alpha_i^2 K(\mathbf{x}_i, \mathbf{x}), \cdots, \underbrace{\sum_{i=1}^{n}\alpha_i^d K(\mathbf{x}_i, \mathbf{x})}_{\text{向第d个投影}}\right]^T \in R^d$$

$$n\lambda\boldsymbol{\alpha} = \mathbf{K}\boldsymbol{\alpha}$$（作特征值分解求α）

$$\boldsymbol{\alpha} = \begin{pmatrix} \alpha_1^1 & \alpha_1^2 & \cdots & \alpha_1^d \\ \alpha_2^1 & \alpha_2^2 & \cdots & \alpha_2^d \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_n^1 & \alpha_n^2 & \cdots & \alpha_n^d \end{pmatrix}$$

第一个最大的特征值对应的特征向量

第d个最大的特征值对应的特征向量

$$\left[\mathbf{v}_1 = \sum_{i=1}^{n}\alpha_i^1\phi(\mathbf{x}_i), \ \mathbf{v}_2 = \sum_{i=1}^{n}\alpha_i^2\phi(\mathbf{x}_i), \quad \ldots, \quad \mathbf{v}_d = \sum_{i=1}^{n}\alpha_i^d\phi(\mathbf{x}_i)\right]$$

d 维特征子空间对应的投影变换矩阵

- **与PCA作对比**
  - 在PCA中，W为样本协方差矩阵的前d个特征值对应的特征向量所构成：$\mathbf{W}=\left[\mathbf{w}_1,\mathbf{w}_2,\cdots,\mathbf{w}_d\right]\in R^{m\times d}$ ，
    - 样本点x的投影： $\mathbf{y}=\mathbf{W}^T\mathbf{x}=\left[\mathbf{w}_1\cdot\mathbf{x},\mathbf{w}_2\cdot\mathbf{x},\cdots,\mathbf{w}_d\cdot\mathbf{x}\right]^T\in R^d$
  - 在KPCA中，**W**为样本在高维特征空间中的协方差矩阵的前d个特征值对应的特征向量所构成：

$$\mathbf{W}=\left[\mathbf{v}_1=\sum_{i=1}^{n}\alpha_i^1\phi(\mathbf{x}_i),\ \mathbf{v}_2=\sum_{i=1}^{n}\alpha_i^2\phi(\mathbf{x}_i),\quad \ldots,\quad \mathbf{v}_d=\sum_{i=1}^{n}\alpha_i^d\phi(\mathbf{x}_i)\right]$$

    - 样本点x的投影：

$$\mathbf{y}=\left[\mathbf{v}_1^T\phi(\mathbf{x}),\quad \mathbf{v}_2^T\phi(\mathbf{x}),\quad \cdots,\quad \mathbf{v}_d^T\phi(\mathbf{x})\right]^T$$

$$=\left[\sum_{i=1}^{n}\alpha_i^1K(\mathbf{x}_i,\mathbf{x}),\sum_{i=1}^{n}\alpha_i^2K(\mathbf{x}_i,\mathbf{x}),\cdots,\sum_{i=1}^{n}\alpha_i^dK(\mathbf{x}_i,\mathbf{x})\right]^T$$

3.9 KPCA

- **LDA**
  - 优化准则

行列式比值

$$\max_{\mathbf{W}\in R^{m\times d},\mathbf{W}^T\mathbf{W}=\mathbf{I}} \frac{\left|\mathbf{W}^T\mathbf{S}_b\mathbf{W}\right|}{\left|\mathbf{W}^T\mathbf{S}_w\mathbf{W}\right|}, \quad \max_{\mathbf{W}\in R^{m\times d},\mathbf{W}^T\mathbf{W}=\mathbf{I}} tr\left(\left(\mathbf{W}^T\mathbf{S}_b\mathbf{W}\right)^{-1}\mathbf{W}^T\mathbf{S}_w\mathbf{W}\right)$$

迹比值

$$\max_{\mathbf{W}\in R^{m\times d},\mathbf{W}^T\mathbf{W}=\mathbf{I}} \frac{tr\left(\mathbf{W}^T\mathbf{S}_b\mathbf{W}\right)}{tr\left(\mathbf{W}^T\mathbf{S}_w\mathbf{W}\right)}$$

迹差值

$$\max_{\mathbf{W}\in R^{m\times d},\mathbf{W}^T\mathbf{W}=\mathbf{I}} tr\left(\mathbf{W}^T(\mathbf{S}_b-\mathbf{S}_w)\mathbf{W}\right)$$

3.8 KPCA

- **LDA重表示（核心）**
  - 记 X=[$\mathbf{x}_1$,$\mathbf{x}_2$, …, $\mathbf{x}_n$]$\in$R$^{n\times m}$，　均值：$\bar{\mathbf{x}} = \frac{1}{n}\left(\sum_{i=1}^{n}\mathbf{x}_i\right)$
  - 零中心化：　$\{\mathbf{x}_1 - \bar{\mathbf{x}}, \mathbf{x}_2 - \bar{\mathbf{x}}, \ldots, \mathbf{x}_n - \bar{\mathbf{x}}\}$
  - **最优的 W** 在零中心化数据所张成的子空间中:
  - 重写 $\mathbf{S}_b$ 和 $\mathbf{S}_w$：$\mathbf{W} = \mathbf{XC\alpha}$,　$\mathbf{C} = \mathbf{I} - \frac{1}{n}\mathbf{e}_n\mathbf{e}_n^T$,　$\boldsymbol{\alpha} \in R^{n\times d}$

$$\mathbf{S}_b = \mathbf{X}\mathbf{L}_b\mathbf{X}^T, \quad \mathbf{S}_w = \mathbf{X}\mathbf{L}_w\mathbf{X}^T$$

拉普拉斯矩阵*

* X. F., He, et al. Face recognition using laplacianfaces. PAMI, 2005, 27(3):328–340.
S., Yan S, et al. Graph Embedding and Extensions: A General Framework for Dimensionality Reduction. PAMI, 2007, 29(1):40–51.

3.9 KLDA

- **LDA重表示**
  - 在子空间中计算：

$$\mathbf{W} = \mathbf{X}\mathbf{C}\boldsymbol{\alpha}$$

类间散度	$\mathbf{W}^T\mathbf{S}_b\mathbf{W} = \boldsymbol{\alpha}^T\mathbf{C}^T\mathbf{X}^T\mathbf{X}\mathbf{L}_b\mathbf{X}^T\mathbf{X}\mathbf{C}\boldsymbol{\alpha},$
类内散度	$\mathbf{W}^T\mathbf{S}_w\mathbf{W} = \boldsymbol{\alpha}^T\mathbf{C}^T\mathbf{X}^T\mathbf{X}\mathbf{L}_w\mathbf{X}^T\mathbf{X}\mathbf{C}\boldsymbol{\alpha},$
子空间	$\mathbf{W}^T\mathbf{W} = \boldsymbol{\alpha}^T\mathbf{C}^T\mathbf{X}^T\mathbf{X}\mathbf{C}\boldsymbol{\alpha}$

  - 应用LDA的优化准则求α

$$\mathbf{y} = \mathbf{W}^T\mathbf{x} = \boldsymbol{\alpha}^T\mathbf{C}^T\mathbf{X}^T\mathbf{x}$$

  - 对新数据：

3.9 KLDA

- **核化**
  - 在子空间中计算：

类间散度	$\mathbf{W}^T\mathbf{S}_b\mathbf{W} = \boldsymbol{\alpha}^T\mathbf{C}^T\mathbf{K}(\mathbf{X},\mathbf{X})\mathbf{L}_b\mathbf{K}(\mathbf{X},\mathbf{X})\mathbf{C}\boldsymbol{\alpha},$
类内散度	$\mathbf{W}^T\mathbf{S}_w\mathbf{W} = \boldsymbol{\alpha}^T\mathbf{C}^T\mathbf{K}(\mathbf{X},\mathbf{X})\mathbf{L}_w\mathbf{K}(\mathbf{X},\mathbf{X})\mathbf{C}\boldsymbol{\alpha},$
子空间	$\mathbf{W}^T\mathbf{W} = \boldsymbol{\alpha}^T\mathbf{C}^T\mathbf{K}(\mathbf{X},\mathbf{X})\mathbf{C}\boldsymbol{\alpha}$

  - 应用LDA的优化准则求$\boldsymbol{\alpha}$
  - 对新数据： $\mathbf{y} = \mathbf{W}^T\mathbf{x} = \boldsymbol{\alpha}^T\mathbf{C}^T\mathbf{K}(\mathbf{X},\mathbf{x})$

3.10 关于核化的一般性理论

- **主要参考文献**
  - Changshui Zhang, Feiping Nie, Shiming Xiang: ***A general kernelization framework for learning algorithms based on kernel PCA***. Neurocomputing 73(4-6): 959-967, 2010

3.10 关于核化的一般性理论

- **满秩PCA**
  - 对于训练数据X，设其中心化的内积矩阵（即协方差矩阵）**C**的秩为 r，如果提取PCA的前 r 个主成分，则称此过程为满秩PCA。
- **满秩KPCA**
  - 对于训练数据X，设其中心化的核矩阵**K**的秩为 r，如果提取KPCA的前 r 个主成分，则称此过程为满秩KPCA。

3.10 关于核化的一般性理论

- **定理：**

如果一个线性算法同时满足如下两个条件：

(1) 算法的**输出仅与内积运算** $\mathbf{x} \cdot \mathbf{x}_i$, $(i = 1, 2, \ldots, n)$ 有关；

（2） 对训练数据的**平移不会改变算法的输出结果**；

则该算法的核化可以通过对数据先做满秩KPCA变换，然后在变换后的数据上直接再做该线算法来实现。

3.10 关于核化的一般性理论

- **举例**
  - **KSVM = KPCA + SVM**
  - **KLDA = KPCA + LDA**
  - **KCCA = KPCA + CCA**
  - **KPLS = KPCA + PLS**
  - **核岭回归 = KPCA + 岭回归**

在实际应用中，对有噪声的数据，采用低秩PCA来做会更好！

Thank All of You!
(Questions?)

向世明
smxiang@nlpr.ia.ac.cn
http://www.escience.cn/people/smxiang
时空数据分析与学习课题组(STDAL)
中科院自动化研究所· 模式识别国家重点实验室