

数据挖掘

第九讲· 分类高级



分类：高级方法

- 1 贝叶斯信念网络
- 2 用后向传播分类
- 3 支持向量机
- 4 其他方法与其他问题
- 5 小结

支持向量机

- 支持向量机 (Support Vector Machines)

- 二分类模型：寻找一个**超平面**来对样本进行分割，分割的原则是**间隔最大化**
- 转化为一个凸二次规划问题来求解
- 由简至繁的模型包括：
 - 当训练样本线性可分时，通过**硬间隔最大化**，学习一个**线性可分支持向量机**；
 - 当训练样本近似线性可分时，通过**软间隔最大化**，学习一个**线性支持向量机**；
 - 当训练样本线性不可分时，通过**核技巧和软间隔最大化**，学习一个**非线性支持向量机**；

分类机

给定训练集 $T = \{(x_1, y_1), \dots, (x_l, y_l)\}$,

其中 $x_i \in X = R^n, y_i \in Y = \{1, -1\}, i = 1, \dots, l$

任务: 对新来的 x , 推断相应的 y , 或者说: 把 R^n 空间分成两部分.

用曲面 $g(x) = 0$ 划分 **(一般分类问题)**

寻找 R^n 上的一个实值函数 $g(x)$, 构造决策函数 $f(x) = \text{sgn}(g(x))$.

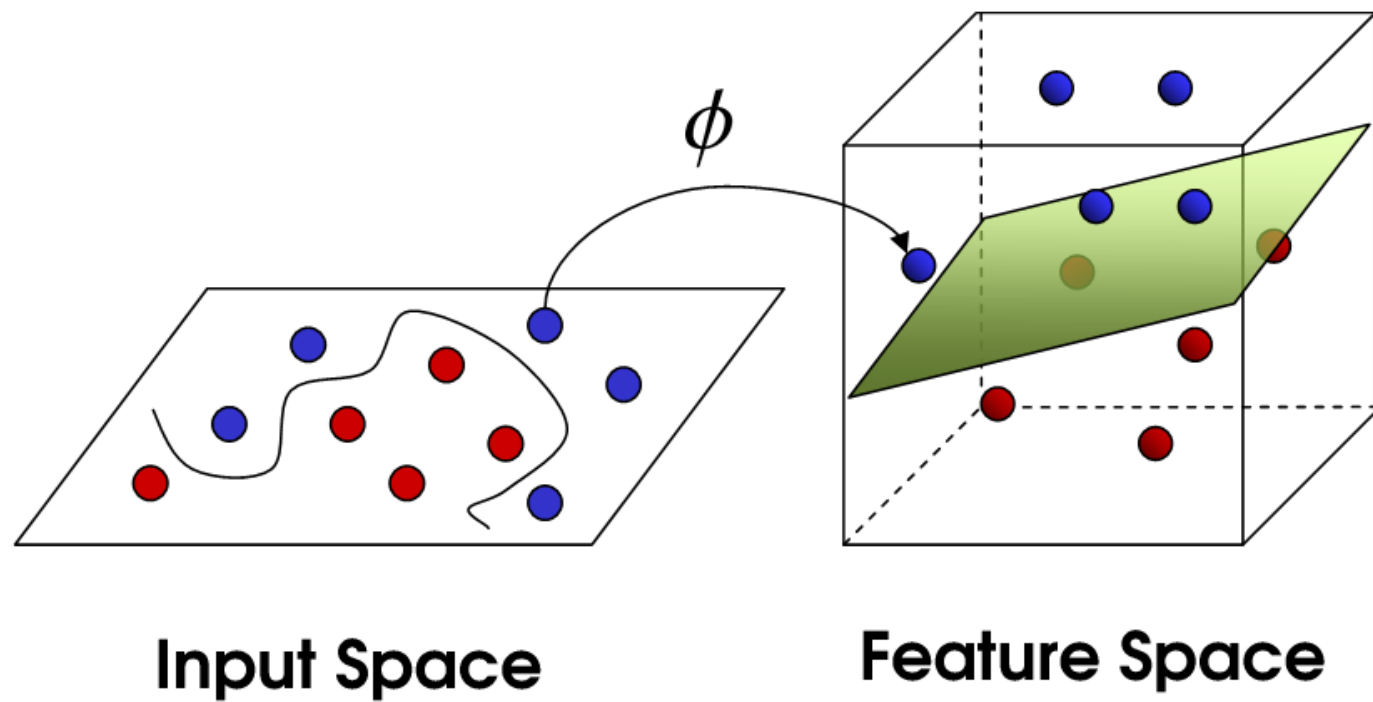
用超平面 $(w \cdot x) + b = 0$ 划分 **(线性分类问题)**

构造决策函数 $f(x) = \text{sgn}((w \cdot x) + b)$

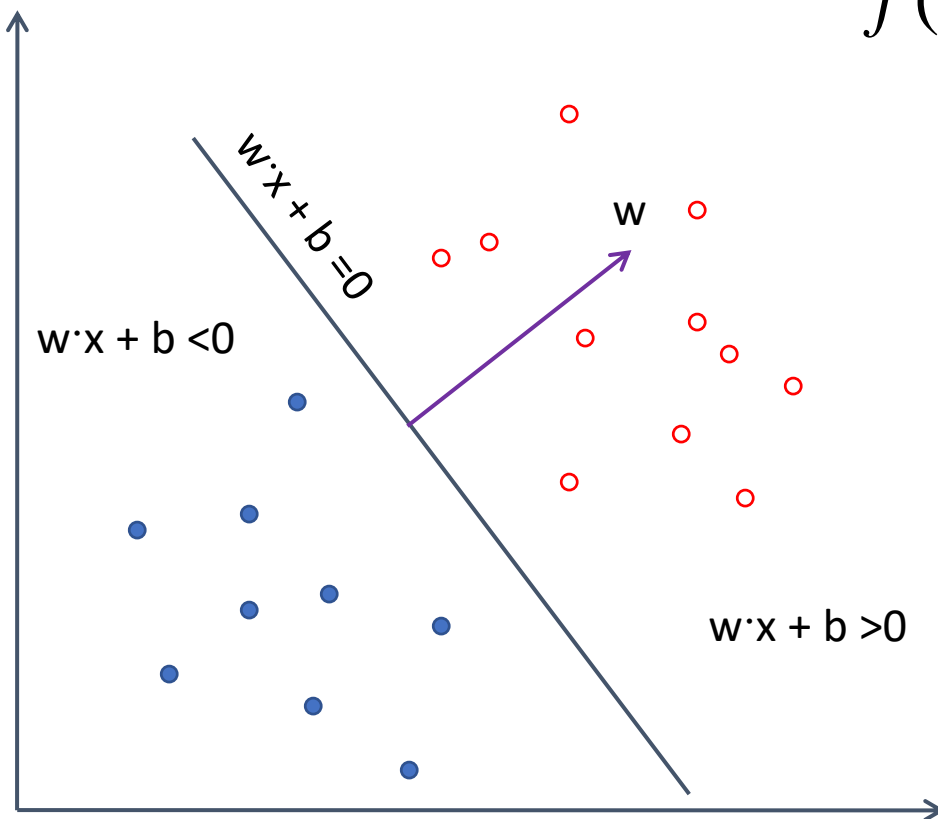
(多类问题)

基本思想与关键概念

基本思想



线性分类机



$$f(x, w, b) = \text{sign}(g(x)) \\ = \text{sign}(w \cdot x + b)$$

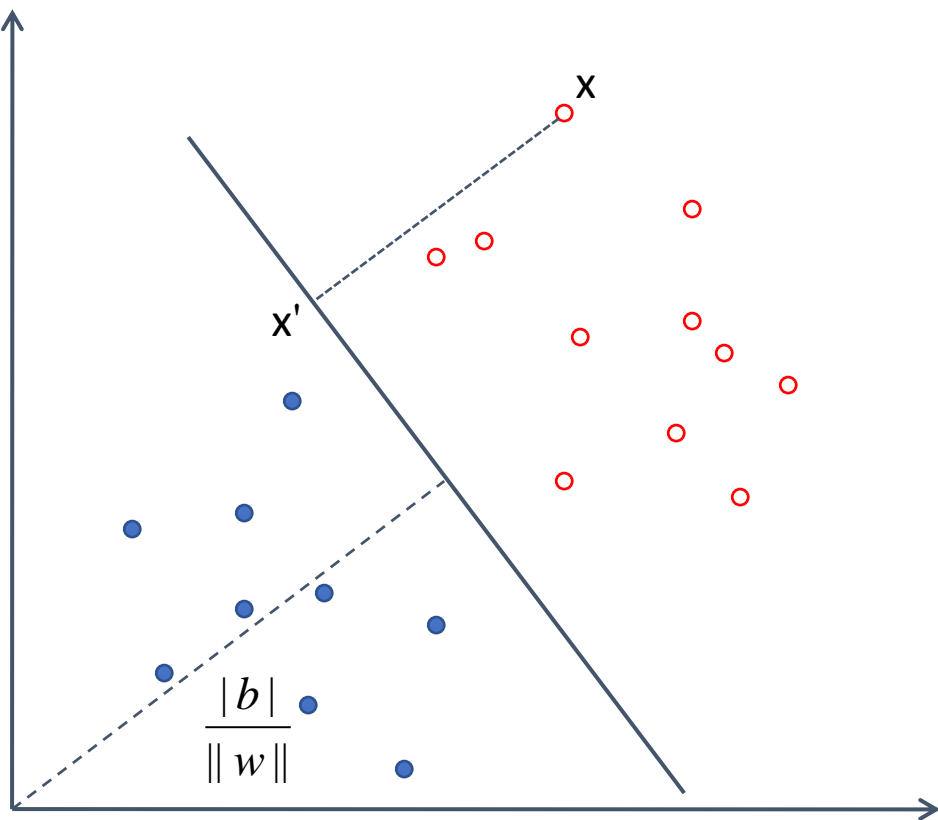
Just in case ...

$$w \cdot x = \sum_{i=1}^n w_i x_i$$

$$w \cdot x_1 + b = w \cdot x_2 + b$$

$$w(x_1 - x_2) = 0$$

到超平面的距离



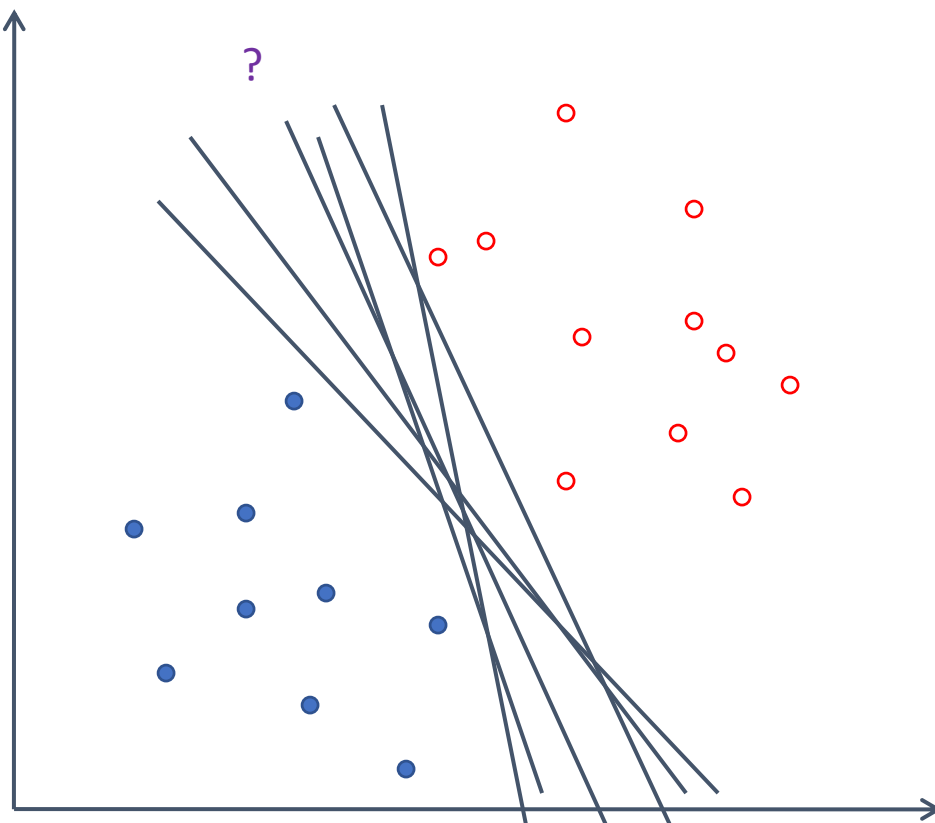
$$g(x) = w \cdot x + b$$

$$x = x' + \lambda w$$

$$\begin{aligned} g(x) &= w(x' + \lambda \cdot w) + b \\ &= w \cdot x' + b + \lambda w \cdot w \\ &= \lambda w \cdot w \end{aligned}$$

$$\begin{aligned} M &= \|x - x'\| = \|\lambda w\| \\ &= \frac{|g(x)| \times \|w\|}{w \cdot w} = \frac{|g(x)|}{\|w\|} \end{aligned}$$

分类机的选择

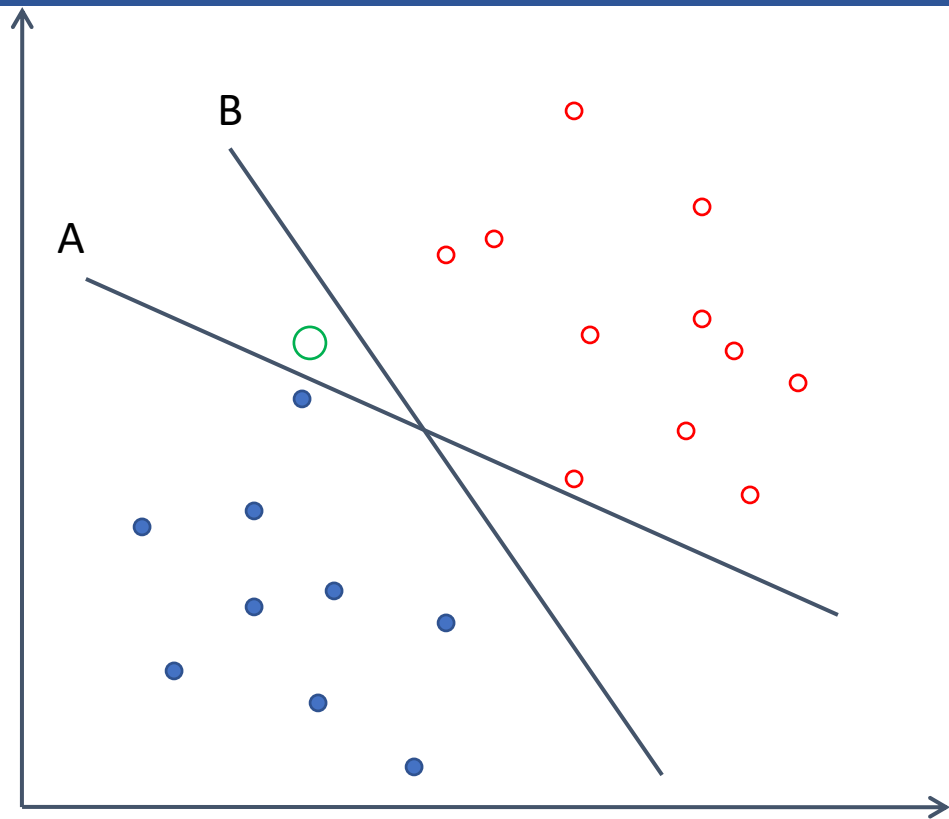


哪个分类效果更好？

同样的训练误差

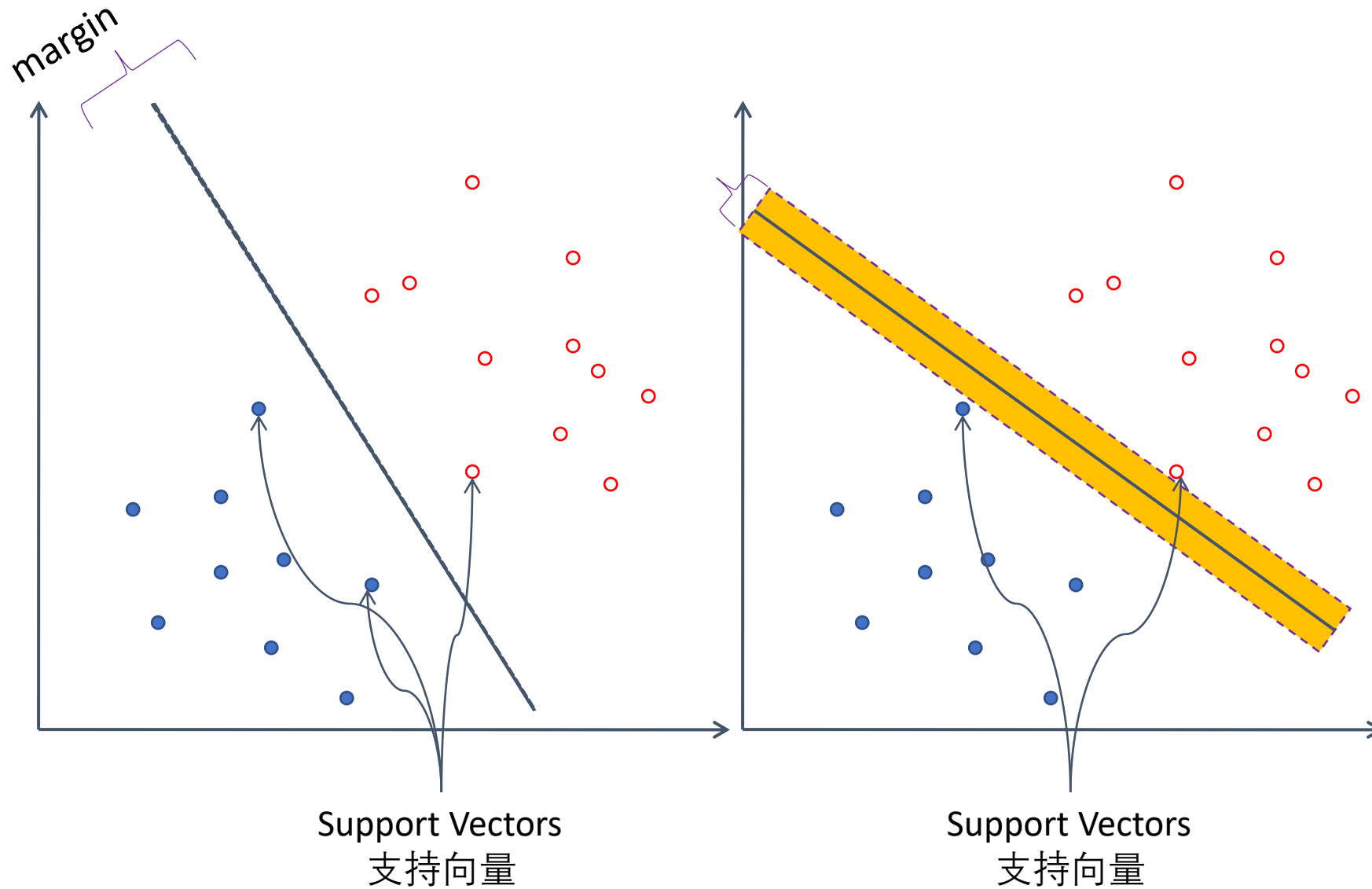
泛化误差如何？

未知样本



B 将点分为两类 (无偏)

间隔 (Margin)

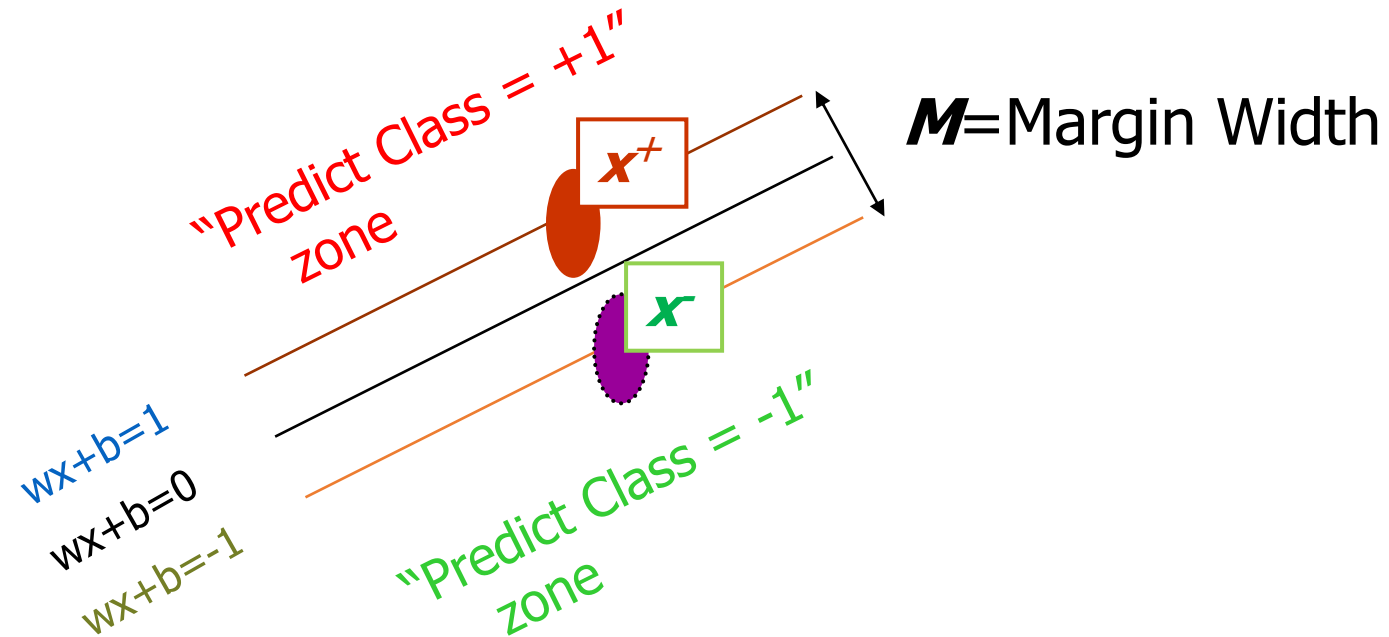


间隔Margins

- 间隔定义：在碰到数据点之前的边界宽度
- 间隔越大越安全？
- 超平面是由少量点组成：
 - 支持向量 Support Vectors
 - 其他的点可以丢弃
- 选择间隔最大的分类机
 - Linear Support Vector Machines (LSVM)
- 怎么界定 间隔？



间隔计算



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



线性支持向量机

目标函数

- 正确分类所有数据点 $w \cdot x_i + b \geq 1 \quad \text{if } y_i = +1$

$$w \cdot x_i + b \leq -1 \quad \text{if } y_i = -1$$

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



- 最大化间隔 $\max M = \frac{2}{\|w\|} \Rightarrow \min \frac{1}{2} w^T w$

- 二次优化问题

- 最小化

$$\Phi(w) = \frac{1}{2} w^T w$$

- 满足

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

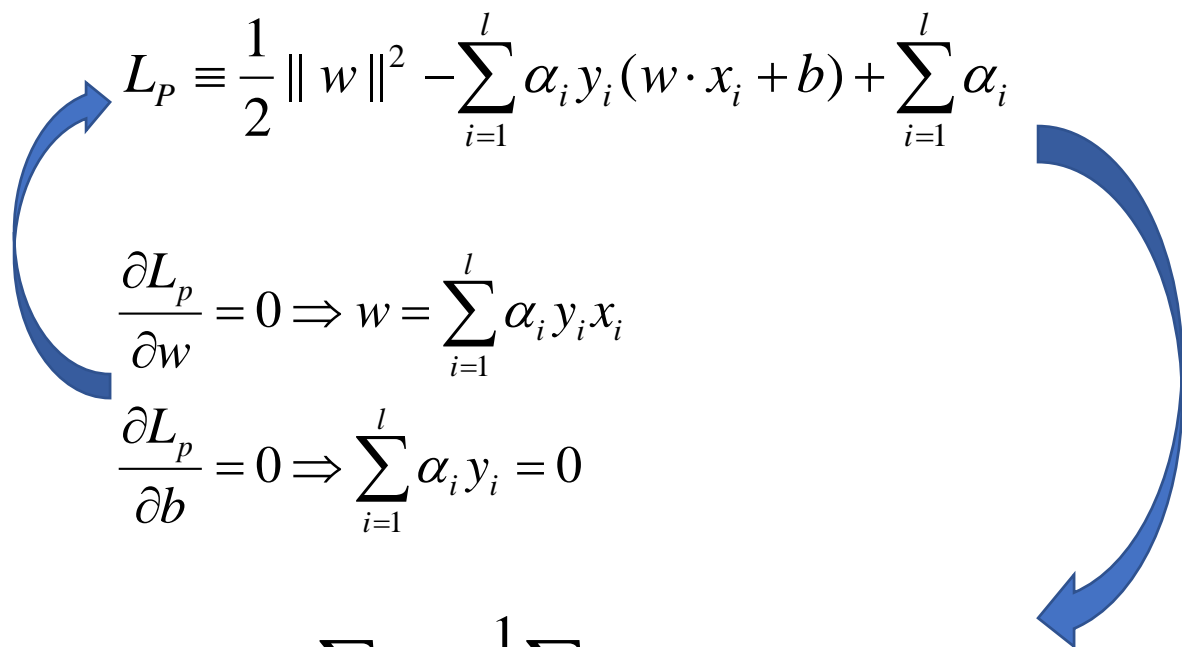
由最大间隔原则，得到线性可分问题的优化问题

$$\begin{array}{ll}\min_{w,b} & \frac{1}{2} \|w\|^2, \\ \text{s.t.} & y_i((w \cdot x_i) + b) \geq 1, i = 1, \dots, l\end{array}$$

求得解 (w^*, b^*) 后，构造决策函数

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

拉格朗日乘子法 (Lagrange Multipliers)


$$L_P \equiv \frac{1}{2} \|w\|^2 - \sum_{i=1}^l \alpha_i y_i (w \cdot x_i + b) + \sum_{i=1}^l \alpha_i$$
$$\frac{\partial L_P}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^l \alpha_i y_i x_i$$
$$\frac{\partial L_P}{\partial b} = 0 \Rightarrow \sum_{i=1}^l \alpha_i y_i = 0$$

对偶问题

$$L_D \equiv \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i \cdot x_j$$
$$\equiv \sum_i \alpha_i - \frac{1}{2} \alpha^T H \alpha \text{ where } H_{ij} = y_i y_j x_i \cdot x_j$$

二次型问题!

$$\text{subject to: } \sum_i \alpha_i y_i = 0 \text{ \& } \alpha_i \geq 0$$

求解 w 和 b

支持向量：参数 α 为正的样本

$$y_s (x_s \cdot w + b) = 1$$

$$y_s \left(\sum_{m \in S} \alpha_m y_m x_m \cdot x_s + b \right) = 1$$

$$y_s^2 \left(\sum_{m \in S} \alpha_m y_m x_m \cdot x_s + b \right) = y_s$$

$$b = y_s - \sum_{m \in S} \alpha_m y_m x_m \cdot x_s$$

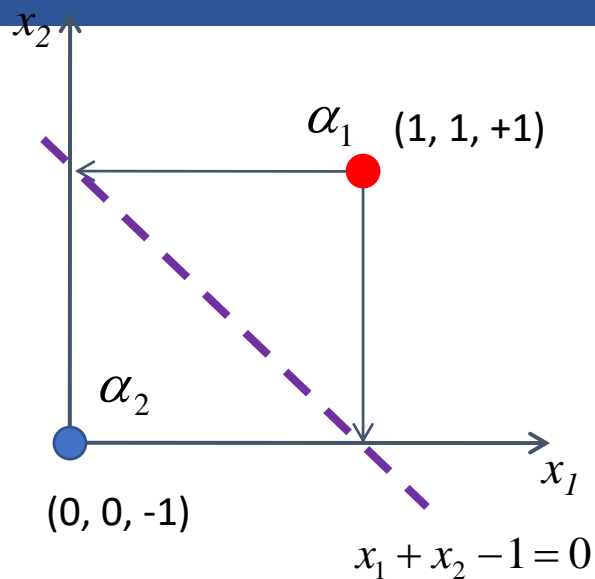
$$b = \frac{1}{N_s} \sum_{s \in S} \left(y_s - \sum_{m \in S} \alpha_m y_m x_m \cdot x_s \right)$$

$$g(x) = \sum_{i=1}^l \alpha_i y_i x_i \cdot x + b$$



内积

例:



$$\sum_{i=1}^2 \alpha_i y_i = 0 \Rightarrow \alpha_1 - \alpha_2 = 0 \Rightarrow \alpha_1 = \alpha_2$$

$$H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} = \begin{bmatrix} y_1 y_1 x_1 \cdot x_1 & y_1 y_2 x_1 \cdot x_2 \\ y_2 y_1 x_2 \cdot x_1 & y_2 y_2 x_2 \cdot x_2 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$$

$$L_D \equiv \sum_{i=1}^2 \alpha_i - \frac{1}{2} [\alpha_1, \alpha_2] H \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = 2\alpha_1 - \alpha_1^2$$

$$w = \sum_{i=1}^2 \alpha_i y_i x_i = 1 \times 1 \times [1, 1] + 1 \times (-1) \times [0, 0] = [1, 1]$$

$$\alpha_1 = 1; \alpha_2 = 1$$

$$b = -wx_1 + 1 = -2 + 1 = -1$$

$$g(x) = wx + b = x_1 + x_2 - 1$$

$$M = \frac{2}{\|w\|} = \frac{2}{\sqrt{2}} = \sqrt{2}$$

算法 (线性支持向量分类机 - C-SVC)

(1) 设已知训练集 $T = \{(x_1, y_1), \dots, (x_l, y_l)\}$;

(2) 选择参数 $C > 0$, 求解最优化问题

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

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

$$0 \leq \alpha_i \leq C, i = 1, \dots, l,$$

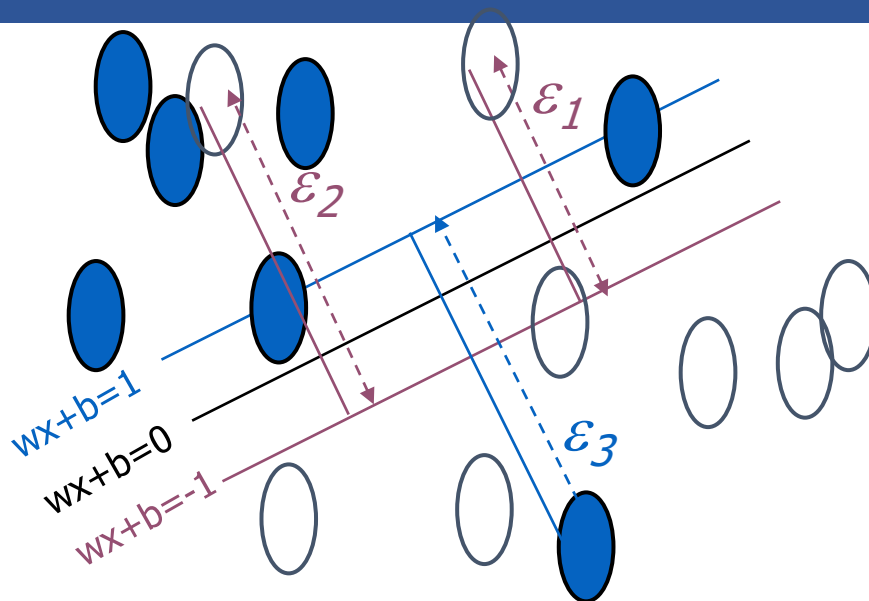
得解 $\alpha^* = (\alpha_1^*, \dots, \alpha_l^*)^T$

(3) 计算 $w^* = \sum_{i=1}^l \alpha_i^* y_i x_i, \quad b^* = \dots$;

(4) 决策函数 $f(x) = \text{sgn}((w^* \cdot x) + b^*) = \text{sgn}((\sum_{i=1}^l \alpha_i^* y_i (x_i \cdot x) + b^*))$

x_i 仅以形式 $(x_i \cdot x_j)$ 出现

软间隔 (Soft Margin)




$$y_i(wx_i + b) - 1 + \xi_i \geq 0$$

$$\Phi(w) = \frac{1}{2} w^t w + C \sum_i \xi_i$$

$$\xi_i \geq 0$$

$$L_P \equiv \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i - \sum_{i=1}^l \alpha_i [y_i(w \cdot x_i + b) - 1 + \xi_i] - \sum_{i=1}^l \mu_i \xi_i$$

软间隔 (Soft Margin)


$$\begin{aligned}\frac{\partial L_P}{\partial w} = 0 &\Rightarrow w = \sum_{i=1}^l \alpha_i y_i x_i \\ \frac{\partial L_P}{\partial b} = 0 &\Rightarrow \sum_{i=1}^l \alpha_i y_i = 0\end{aligned}\quad \left. \vphantom{\begin{aligned}\frac{\partial L_P}{\partial w} = 0 \\ \frac{\partial L_P}{\partial b} = 0\end{aligned}} \right\} \text{同前}$$
$$\frac{\partial L_P}{\partial \xi_i} = 0 \Rightarrow C = \alpha_i + \mu_i$$
$$L_P \equiv \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i - \sum_{i=1}^l \alpha_i [y_i (w \cdot x_i + b) - 1 + \xi_i] - \sum_{i=1}^l \mu_i \xi_i$$

$$L_D \equiv \sum_i \alpha_i - \frac{1}{2} \alpha^T H \alpha \quad s.t. \ 0 \leq \alpha_i \leq C \quad and \ \sum_i \alpha_i y_i = 0$$

近似线性可分分类机

近似线性可分问题的优化问题：

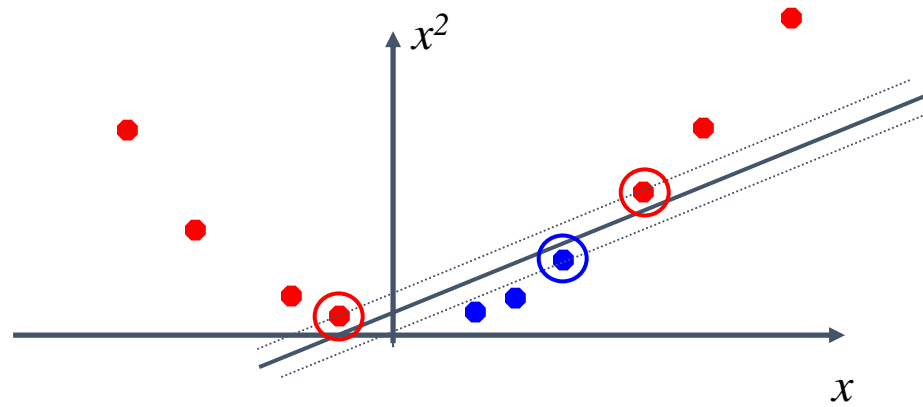
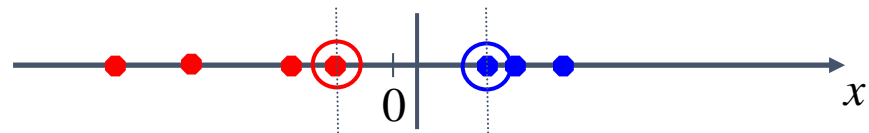
$$\begin{aligned} \min_{w, b, \xi} \quad & \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i, \\ \text{s.t.} \quad & y_i((w \cdot x_i) + b) \geq 1 - \xi_i, \quad i = 1, \dots, l, \\ & \xi_i \geq 0, \quad i = 1, \dots, l \end{aligned}$$

其中 $C > 0$ 是一个惩罚参数.

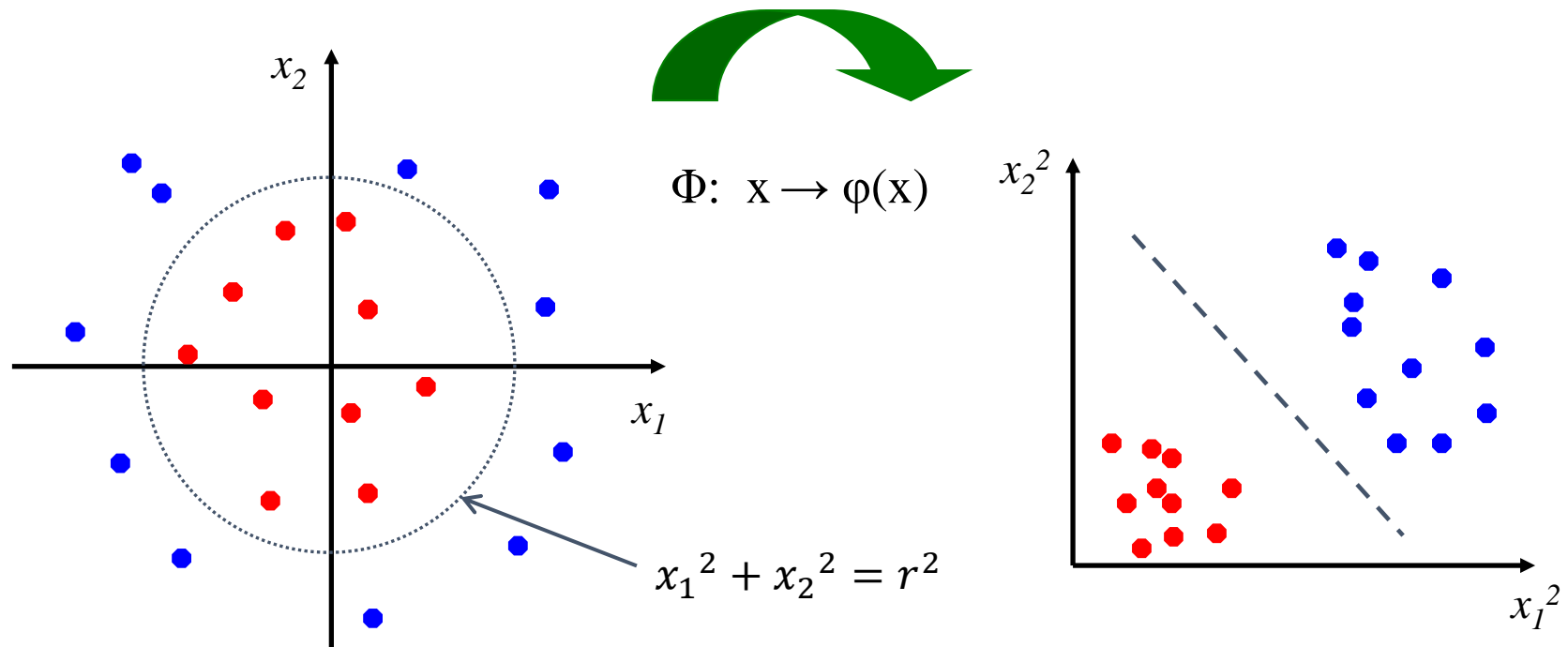
求得解 (w^*, b^*, ξ^*) 后, 构造决策函数 $f(x) = \text{sgn}((w^* \cdot x) + b^*)$

非线性支持向量机

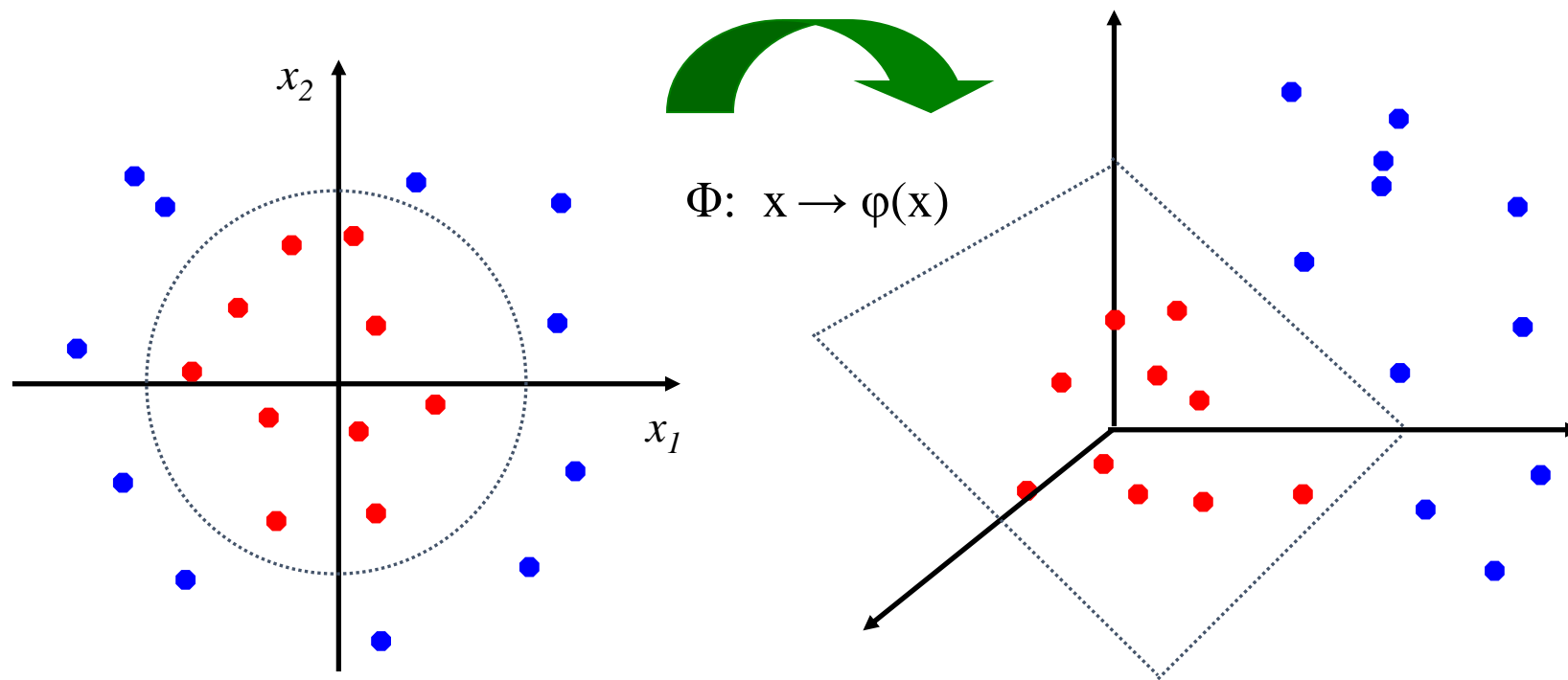
非线性 SVM



特征空间



特征空间



二次基函数

$$\Phi(x) = \left[\begin{array}{c} 1 \\ \sqrt{2}x_1 \\ \sqrt{2}x_2 \\ \vdots \\ \sqrt{2}x_m \\ x_1^2 \\ x_2^2 \\ \vdots \\ x_m^2 \\ \sqrt{2}x_1x_2 \\ \sqrt{2}x_1x_3 \\ \vdots \\ \sqrt{2}x_1x_m \\ \sqrt{2}x_2x_3 \\ \vdots \\ \sqrt{2}x_2x_m \\ \vdots \\ \sqrt{2}x_{m-1}x_m \end{array} \right] \left\{ \begin{array}{l} \text{常数项} \\ \text{线性项} \\ \text{纯二次项} \\ \text{二次交叉项} \end{array} \right.$$

项的总数

$$C_{m+2}^2 = \frac{(m+2)(m+1)}{2} \approx \frac{m^2}{2}$$

计算 $\Phi(x_i) \cdot \Phi(x_j)$

$$\Phi(a) \cdot \Phi(b) = \begin{bmatrix} 1 \\ \sqrt{2}a_1 \\ \sqrt{2}a_2 \\ \vdots \\ \sqrt{2}a_m \\ a_1^2 \\ a_2^2 \\ \vdots \\ a_m^2 \\ \sqrt{2}a_1a_2 \\ \sqrt{2}a_1a_3 \\ \vdots \\ \sqrt{2}a_1a_m \\ \sqrt{2}a_2a_3 \\ \vdots \\ \sqrt{2}a_2a_m \\ \vdots \\ \sqrt{2}a_{m-1}a_m \end{bmatrix} \cdot \begin{bmatrix} 1 \\ \sqrt{2}b_1 \\ \sqrt{2}b_2 \\ \vdots \\ \sqrt{2}b_m \\ b_1^2 \\ b_2^2 \\ \vdots \\ b_m^2 \\ \sqrt{2}b_1b_2 \\ \sqrt{2}b_1b_3 \\ \vdots \\ \sqrt{2}b_1b_m \\ \sqrt{2}b_2b_3 \\ \vdots \\ \sqrt{2}b_2b_m \\ \vdots \\ \sqrt{2}b_{m-1}b_m \end{bmatrix} = \begin{matrix} \mathbf{1} \\ \sum_{i=1}^m 2a_i b_i \\ \\ \sum_{i=1}^m a_i^2 b_i^2 \\ \\ \sum_{i=1}^{m-1} \sum_{j=i+1}^m 2a_i a_j b_i b_j \end{matrix}$$

$x_i \cdot x_j \Rightarrow \Phi(x_i) \cdot \Phi(x_j)$

计算 $\Phi(x_i) \cdot \Phi(x_j)$

$$\Phi(a) \cdot \Phi(b) = 1 + 2 \sum_{i=1}^m a_i b_i + \sum_{i=1}^m a_i^2 b_i^2 + \sum_{i=1}^{m-1} \sum_{j=i+1}^m 2a_i a_j b_i b_j$$

$$(a \cdot b + 1)^2 = (a \cdot b)^2 + 2a \cdot b + 1 = \left(\sum_{i=1}^m a_i b_i \right)^2 + 2 \sum_{i=1}^m a_i b_i + 1$$

$$= \sum_{i=1}^m \sum_{j=1}^m a_i b_i a_j b_j + 2 \sum_{i=1}^m a_i b_i + 1$$

$$= \sum_{i=1}^m (a_i b_i)^2 + 2 \sum_{i=1}^{m-1} \sum_{j=i+1}^m a_i b_i a_j b_j + 2 \sum_{i=1}^m a_i b_i + 1$$

$$K(a, b) = (a \cdot b + 1)^2 = \Phi(a) \cdot \Phi(b)$$

$O(m)$

$O(m^2)$

核函数思想

- 支持向量机通过某非线性变换 $\phi(x)$ ，将输入空间映射到高维特征空间。
 - 如果支持向量机的求解只用到内积运算，而在低维输入空间又存在某个函数 $K(x, x')$ ，它恰好等于在高维空间中这个内积，即 $K(x, x') = \langle \phi(x) \cdot \phi(x') \rangle$ 。
 - 支持向量机就不用计算复杂的非线性变换，而由这个函数 $K(x, x')$ 直接得到非线性变换的内积，大大简化了计算。
 - 这样的函数 $K(x, x')$ 称为核函数。

核函数

定义（核函数） 设 $\mathcal{X} \subseteq R^n$ ，称定义在 $\mathcal{X} \times \mathcal{X}$ 上的函数 $K(x, x')$ 为核函数，如果存在着从 \mathcal{X} 到某个Hilbert空间 H 的映射

$$\Phi: \begin{cases} \mathcal{X} \rightarrow H \\ x \rightarrow \Phi(x) \end{cases}$$

使得 $K(x, x') = (\Phi(x) \cdot \Phi(x'))$

其中 (\cdot) 表示 H 中的内积.

常用核函数:

Gauss径向基核 $K(x, x') = \exp(-\|x - x'\|^2 / \sigma^2)$

多项式核 $K(x, x') = ((x \cdot x') + c)^d \quad c \geq 0, d > 0$

傅里叶核 $K(x, x') = \frac{1 - q^2}{2(1 - 2q \cos(x - x') + q^2)}, \quad 0 < q < 1$

Sigmoid核 $K(x, x') = \tanh(\kappa(x \cdot x') + \nu)$

计算 w & b

$$w = \sum_{i=1}^l \alpha_i y_i \Phi(x_i)$$

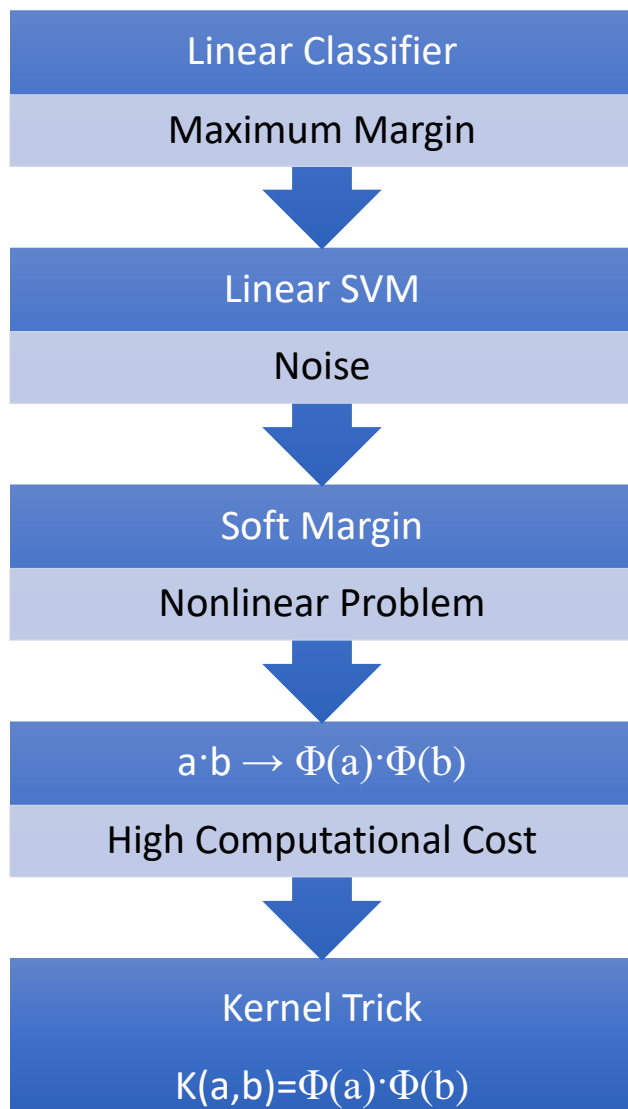
$$w \cdot \Phi(x_j) = \sum_{i=1}^l \alpha_i y_i \Phi(x_i) \cdot \Phi(x_j) = \sum_{i=1}^l \alpha_i y_i K(x_i, x_j)$$

$$b = \frac{1}{N_s} \sum_{s \in S} (y_s - \sum_{m \in S} \alpha_m y_m \Phi(x_m) \cdot \Phi(x_s)) = \frac{1}{N_s} \sum_{s \in S} (y_s - \sum_{m \in S} \alpha_m y_m K(x_m, x_s))$$

$$g(x) = \sum_{i=1}^l \alpha_i y_i K(x_i, x) + b \quad \longleftrightarrow \quad g(x) = w \cdot x + b = \sum_{i=1}^l \alpha_i y_i x_i \cdot x + b$$

支持向量机演变历史

SVM 路线图



SVM 创始人



弗拉基米尔·万普尼克 (Vladimir Naumovich Vapnik)

Professor of Columbia, Fellow of NEC Labs America,
在 nec-labs.com 的电子邮件经过验证

[machine learning](#) [statistics](#) [computer science](#)

[创建我的个人资料](#)

引用次数 [查看全部](#)

	引用次数	年份
The Nature of Statistical Learning Theory V Vapnik Data mining and knowledge discovery	101987 *	1995
Support-vector networks C Cortes, V Vapnik Machine learning 20, 273-297	59850	1995
A training algorithm for optimal margin classifiers BE Boser, IM Guyon, VN Vapnik Proceedings of the fifth annual workshop on Computational learning theory ...	15940	1992
Backpropagation applied to handwritten zip code recognition Y LeCun, B Boser, JS Denker, D Henderson, RE Howard, W Hubbard, ... Neural computation 4 (4) 544-554	14133	1989



sklearn中SVM的算法库

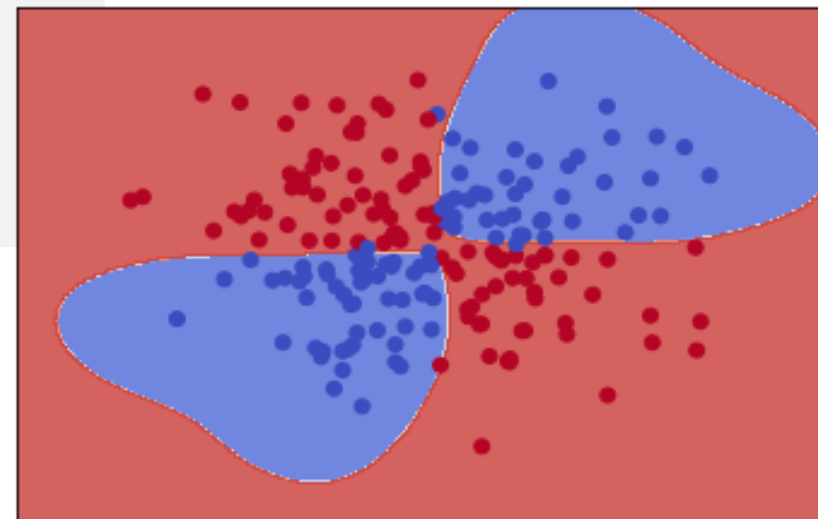
- 分类算法库： 主要包含 LinearSVC, NuSVC和SVC 三个类
- 回归算法库， 包含SVR, NuSVR 和 LinearSVR 三个类
- 相关模块都包裹在sklearn.svm模块中。

```
from sklearn import svm  
  
from mpl_toolkits.mplot3d import Axes3D  
  
from sklearn.model_selection import GridSearchCV
```

sklearn中SVM的算法库调用举例

```
grid = GridSearchCV(svm.SVC(), param_grid={"C": [0.1, 1, 10],  
"gamma": [1, 0.1, 0.01]}, cv=4)  
  
grid.fit(X_xor, y_xor)  
  
print("The best parameters are %s with a score of %0.2f"  
      % (grid.best_params_, grid.best_score_))
```

The best parameters are {'C': 1,
'gamma': 1} with a score of 0.94



gamma=1 C=1

SVM总结

- SVM的主要思想可以概括为两点：
 - 针对线性可分情况进行分析；对于线性不可分的情况，通过使用非线性映射算法将低维输入空间线性不可分的样本转化为高维特征空间使其线性可分，从而使得高维特征空间采用线性算法对样本的非线性特征进行线性分析成为可能。
 - 它基于结构风险最小化理论之上在特征空间中构建最优超平面，使得学习器得到全局最优化，并且在整个样本空间的期望以某个概率满足一定上界。
- 要点与难点：最大间隔，支持向量，凸优化问题，核函数

SVM应用

- **文本和超文本分类：**可以显著减少对标准感应和转换设置中标记的训练实例的需求
- **图像分类：**搜索精度要比传统的查询优化方案高得多
- **生物科学和其他科学领域：**对高达90%正确分类的化合物进行蛋白质分类
- **识别手写字符**

SVM Related Links

- SVM Websites:
 - <http://www.kernel-machines.org/>
 - <http://www.support-vector-machines.org/>
- Representative implementations
 - **LIBSVM**: an efficient implementation of SVM, multi-class classifications, nu-SVM, one-class SVM, including also various interfaces with java, python, etc.
 - **SVM-light**: simpler but performance is not better than LIBSVM, support only binary classification and only in C
 - **SVM-torch**: another recent implementation also written in C

分类：高级方法

- 1 贝叶斯信念网络
- 2 用后向传播分类
- 3 支持向量机
- 4 其他方法与其他问题
- 5 小结

懒惰与急切学习

- 懒惰与急切学习

- 懒惰学习（**Lazy learning**，例如，基于实例的学习）：只需存储训练数据（或仅进行少量处理）并等待，直到得到一个测试元组
- 急切学习（**Eager learning**，如上面讨论的方法）：给定一组训练元组，在接收新的（例如，测试）数据进行分类之前，构造一个分类模型

- 懒惰：训练时间少，预测时间多

- 准确度

- **Lazy learning**：使用许多局部线性函数对目标函数进行隐式全局逼近，因此它有效地利用了更丰富的假设空间
- **Eager learning**：必须有一个覆盖整个实例空间的假设

惰性学习法

- K-最近邻分类(k-Nearest Neighbor, KNN)
 - 如果一个样本在特征空间中的k个最相似(即特征空间中最邻近)的样本中的大多数属于某一个类别, 则该样本也属于这个类别。
- 基于案例的推理 (Case-Based Reasoning, CBR)
 - 根据检索以往的解决问题的经验来解决新问题, 将以往问题的解决方案, 经过修改应用到当前的新情况。
- 遗传算法 (Genetic Algorithm)
- 粗糙集方法(Rough Set Approach)
- 模糊集方法(Fuzzy Logic)

其他问题

- 多类分类
- 半监督分类(Semi-Supervised Learning, SSL)
 - 是监督学习与[无监督学习](#)相结合的一种学习方法。
 - 半监督学习使用大量的未标记数据，以及同时使用标记数据，来进行模式识别工作。
 - 当使用半监督学习时，将会要求尽量少的人员来从事工作，同时，又能够带来比较高的准确性，因此，半监督学习目前正越来越受到人们的重视。
- 主动学习 (Active Learning)
- 迁移学习 (Transfer Learning)

分类：高级方法

- 1 贝叶斯信念网络
- 2 用后向传播分类
- 3 支持向量机
- 4 其他方法与其他问题
- 5 小结

小结

贝叶斯信念
网络分类

基于后验概率的
贝叶斯定理

用后向传播
分类

支持向量机

- 惰性学习法(或从近邻学习)
 - k-最近邻分类、基于案例的推理
- 其他分类方法
 - 遗传算法、粗糙集方法、模糊集方法
- 关于分类的其他问题
 - 多类分类、半监督分类
 - 主动学习、迁移学习

