# 机器学习
# Machine Learning

北京航空航天大学计算机学院
School of Computer Science and Engineering, Beihang University
黄 迪　张永飞　陈佳鑫

2023年秋季学期
Fall 2023

部分内容来源于C. Bishop和A. NG等人的课程以及互联网资源

# 生成式模型和判别式模型

- **生成式模型 (Generative Model)**

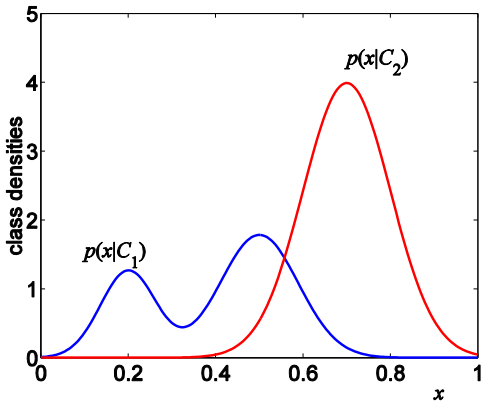
分别对各类的**类条件密度** $p(\mathbf{x}|C_k)$ 和**先验概率** $p(C_k)$ 进行建模，之后利用贝叶斯定理计算**后验概率**

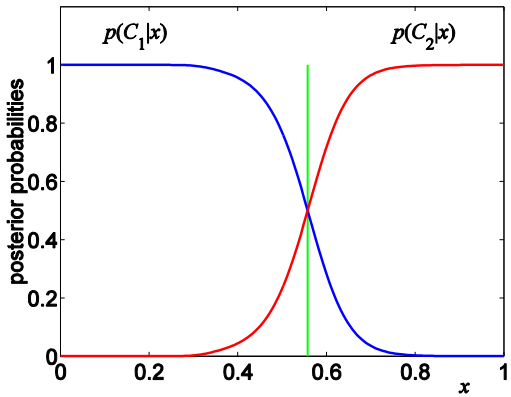
$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{\sum_j p(\mathbf{x}|C_j)p(C_j)}$$

或者直接对**联合分布** $p(\mathbf{x}, C_k)$ 建模得到后验概率

- **判别式模型 (Discriminative Model)**

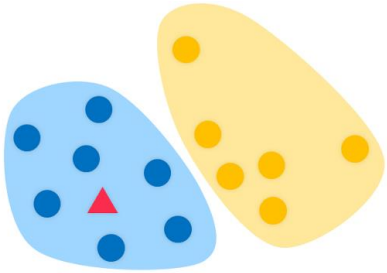
直接对**后验概率** $p(C_k|\mathbf{x})$ 建模



# 生成式模型和判别式模型

## ● 生成式模型



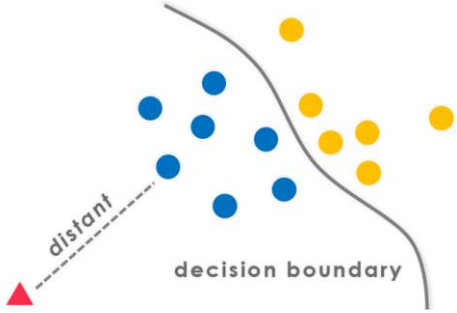
**优点：**

- 信息丰富
- 单类问题灵活性强
- 增量学习
- 合成缺失数据

**缺点：**

- 学习过程复杂
- 为分布牺牲分类性能

## ● 判别式模型




**优点：**

- 类间差异清晰
- 分类边界灵活
- 学习简单
- 性能较好

**缺点：**

- 不能反应数据特性
- 需要全部数据进行学习

**由生成模型可以得到判别模型，**
**但由判别模型得不到生成模型。**

# 生成式模型和判别式模型

## ● 生成式模型

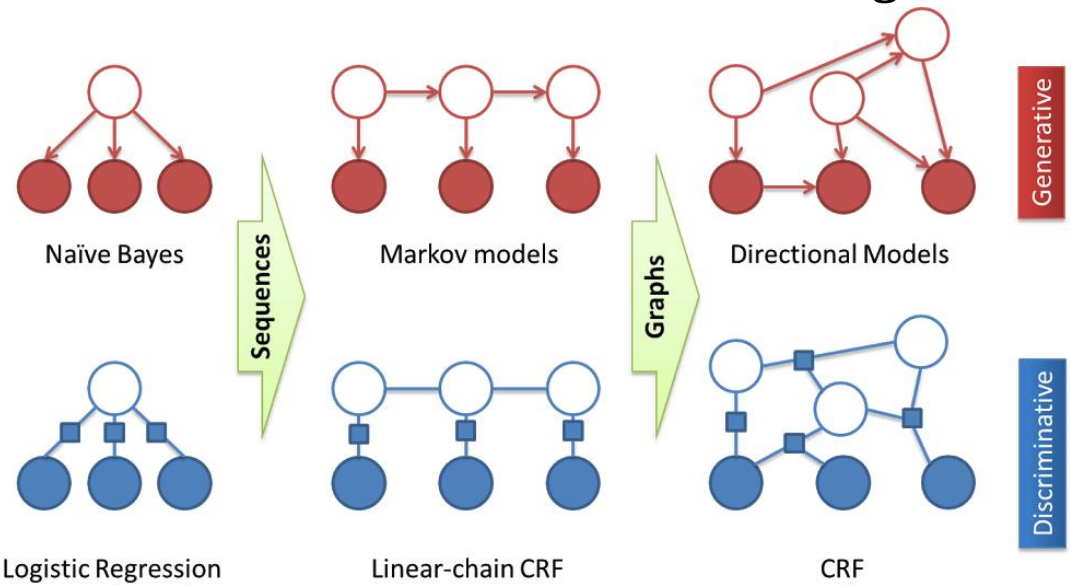
**代表算法：**

- Naive Bayes
- Mixtures of Gaussians
- Hidden Markov Models
- Bayesian Networks
- Deep Belief Network

## ● 判别式模型

**代表算法：**

- Linear & Logistic Regression
- Support Vector Machine
- Nearest Neighbor
- Conditional Random Fields
- Boosting



# 第5章：支持向量机

## Chapter 5: Support Vector Machine (SVM)

# 概述

- C. Cortes和V. Vapnik (1995年提出)

支持向量机是基于统计学习理论(Statistical Learning Theory, SLT)发展起来的一种机器学习的方法。

统计学习理论主要创立者是Vladimir N. Vapnik。

AT&T

# 概述

- **Vladimir N. Vapnik**

**1936年** 出生于苏联

**1958年** 乌兹别克国立大学 硕士
**1964年** 莫斯科控制科学学院 博士

**1964-1990年** 莫斯科控制科学学院
曾担任计算机科学与研究系主任
**1991-2001年** 美国AT&T贝尔实验室
发明支持向量机理论
**2002-2014年** NEC实验室(美国)
从事机器学习研究
**2014-2016年** 美国Facebook公司
从事人工智能研究
**2016年至今** 美国Vencore实验室
继续研究工作

**1995年和2003年，他分别被伦敦大学皇家霍洛威学院和美国哥伦比亚大学聘为计算机专业的教授。**
**2006年，他成为美国国家工程院院士。**

# 概述

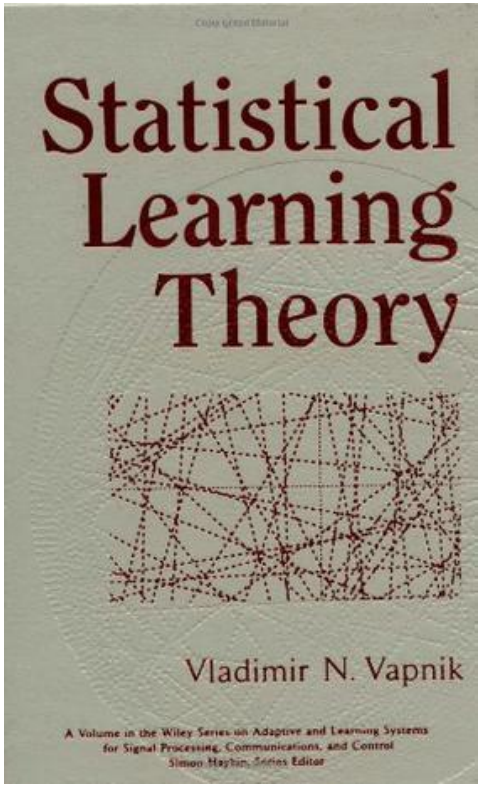
- V. Vapnik对于统计机器学习的贡献

**1968年，Vapnik和Chervonenkis提出了VC熵和VC维的概念，这些是统计学习理论的核心概念。同时，他们发现了泛函空间的大数定理，得到了关于收敛速度的非渐进界的主要结论。**

**1974年，Vapnik和Chervonenkis提出了结构风险最小化归纳原则。**

**1989年，Vapnik和Chervonenkis发现了经验风险最小化归纳原则和最大似然方法一致性的充分必要条件，完成了对经验风险最小化归纳推理的分析。**

**90年代中期，有限样本情况下的机器学习理论研究逐渐成熟起来，形成了较完善的理论体系—统计学习理论。**



# 概述

- 支持向量机的发展

1963年，Vapnik在解决模式识别问题时提出了支持向量方法，这种方法从训练集中选择一组特征子集，使得对特征子集的划分等价于对整个数据集的划分，这组特征子集就被称为支持向量(SV)。

1971年，Kimeldorf提出使用线性不等约束重新构造SV的核空间，解决了一部分线性不可分问题。

1990年，Grace、Boser和Vapnik等人开始对SVM进行研究。

1995年，Vapnik的书《The Nature of Statistical Learning Theory》出版，详细叙述了SVM理论，同时也标志着统计学习理论体系已经走向成熟。

1999年， IEEE Trans. on Neural Network (IEEE T-NN) 为统计学习理论出版了专刊，MIT出版了《Advances in Kernel Method》，使SVM理论的研究与应用推向了一个高潮。

近年来，SVM的研究主要集中在对SVM本身性质的研究和完善以及加大SVM应用研究的深度和广度两方面。

Support-vector networks
C Cortes, V Vapnik
Machine learning 20 (3), 273-297

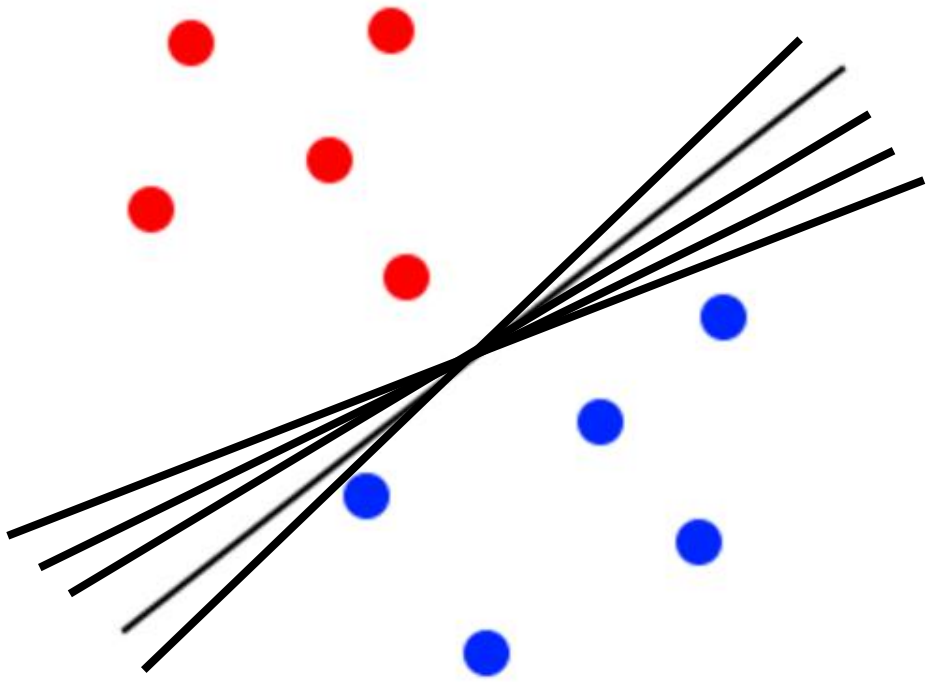
35944 1995

# 线性分类模型

- **两类样本的线性分类问题**

$$y(x) = w^T x + b$$

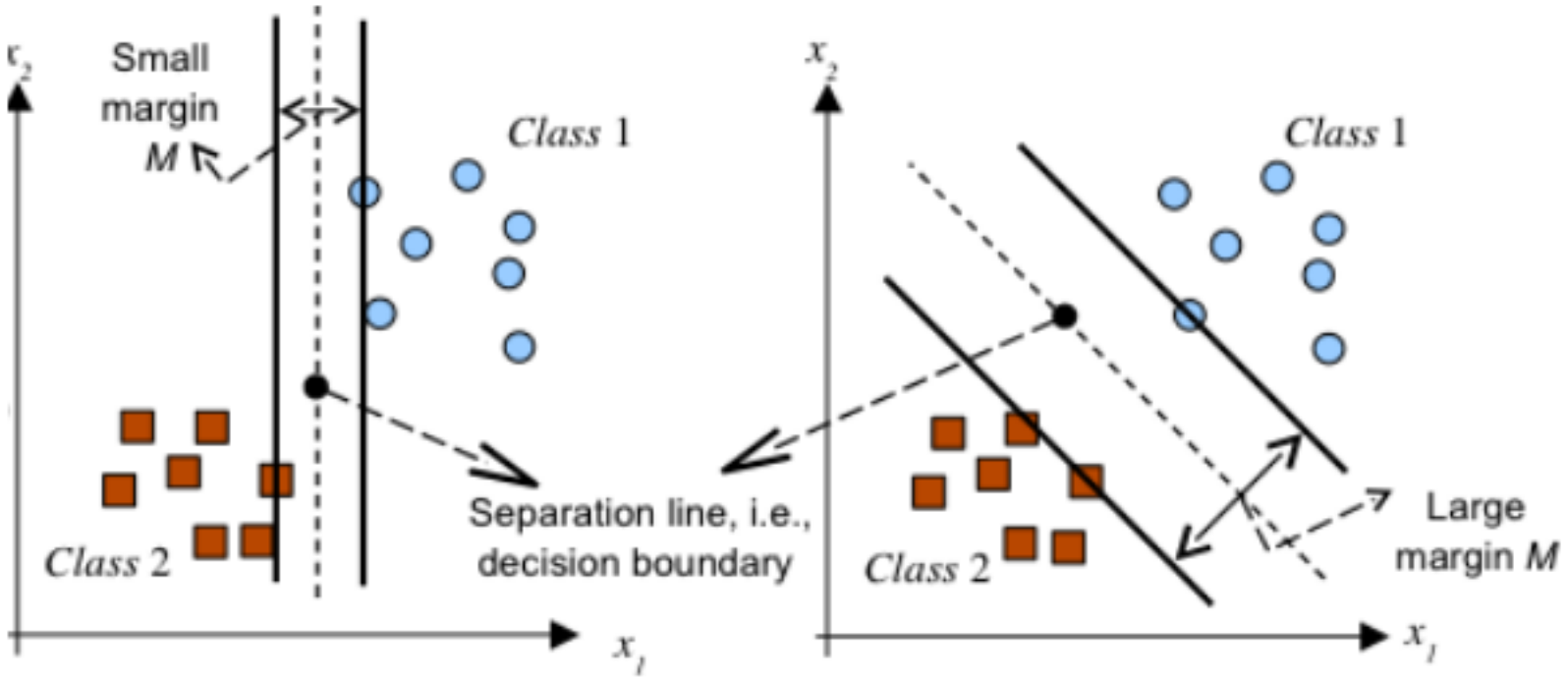
$$y(x, w) = f(\sum_{j=1}^{M} w_j x_j)$$



# 支持向量机

- SVM从线性可分情况下的**最优分类面**发展而来。

最优分类面就是要求分类线**不但能将两类正确分开**(训练错误率为0)，且使**分类间隔**最大。SVM考虑寻找一个满足分类要求的超平面，并且**使训练集中的点距离分类面尽可能的远**，也就是寻找一个分类面**使它两侧的空白区域(Margin)最大**。



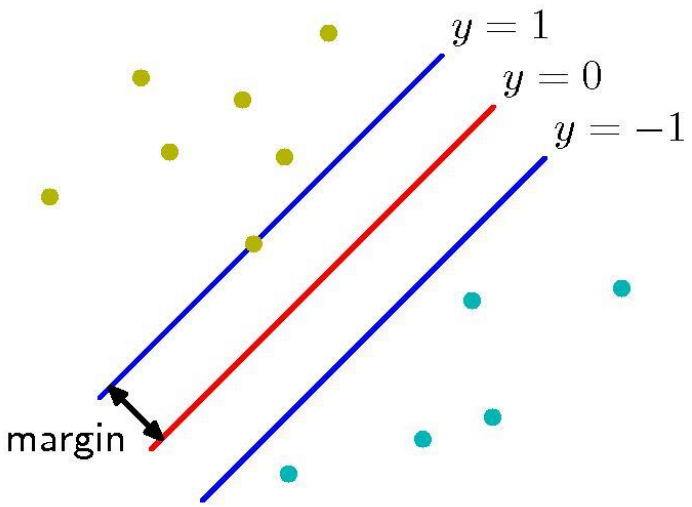
# 支持向量机

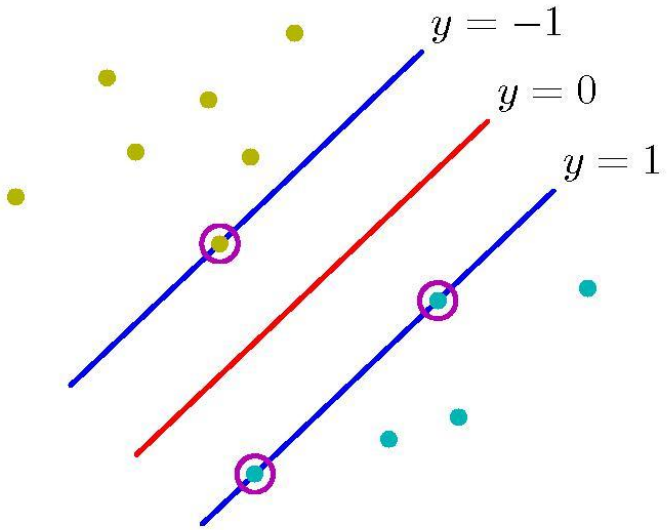
- **线性支持向量机**

**样本集** $\{x_n, t_n\}, n = 1, 2, ..., N,\ x_n \in \mathcal{R}^d; t_n \in \{-1, 1\}$

**分类器** $y(x) = w^T x + b$

$$t_n = \begin{cases} 1, y(x_n) > 0 \text{ if } x_i \in w_1 \\ -1, y(x_n) < 0 \text{ if } x_i \in w_2 \end{cases} \quad \Rightarrow \quad t_n y(x_n) > 0$$



# 支持向量机

- **线性支持向量机**

**样本集任意一点$x_n$到分类面(满足$t_n y(x_n) > 0$)的距离**

$$\frac{t_n y(x_n)}{\|w\|} = \frac{t_n(w^T x_n + b)}{\|w\|}$$

**优化*w*和*b*使Margin最大**

$$\arg\max_{w,b}\{\frac{1}{\|w\|}\min_n[t_n(w^T x_n + b)]\}$$

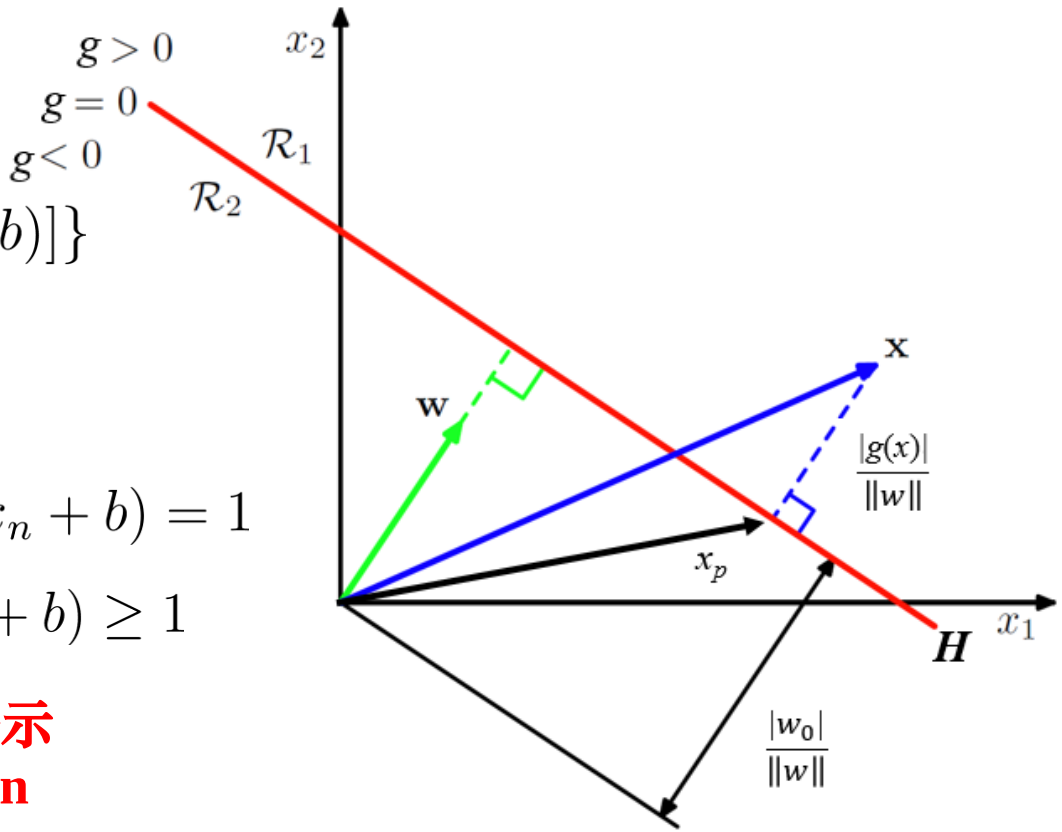
**求解复杂**

$$w \to kw, b \to kb$$

**对于离超平面最近的点** $t_n(w^T x_n + b) = 1$

**那么对于所有点满足** $t_n(w^T x_n + b) \geq 1$

**对于决策超平面的标准表示**
**Canonical Representation**



# 支持向量机

- **线性支持向量机**

**问题转化为最大化** $\|w\|^{-1}$**, 等价于** $\arg\min_{w,b} \frac{1}{2}\|w\|^2$ **二次规划问题**

$$s.t.\ t_n(w^T x_n + b) \geq 1$$

**拉格朗日乘子法** $L(w,b,a) = \frac{1}{2}\|w\|^2 - \sum_{n=1}^{N} a_n\{t_n(w^T x_n + b) - 1\}, a_n \geq 0$

**分别对变量求导** $\frac{\partial L(w,b,a)}{\partial w} = w - \sum_{n=1}^{N} a_n t_n x_n = 0$

$$\frac{\partial L(w,b,a)}{\partial b} = \sum_{n=1}^{N} a_n t_n = 0$$

**代入*L*得到对偶形式:**

$$\tilde{L}(a) = \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m t_n t_m x_n^T x_m$$ **二次规划问题**

$$w.r.t.\ a_n \geq 0, n = 1, ..., N, \sum_{n=1}^{N} a_n t_n = 0$$

# 对偶问题

- **拉格朗日对偶性**

$$\min_w \ f(w)$$

**存在等式约束的函数极值问题** $s.t.\ h_i(w) = 0, i = 1, ..., l.$

$$\Rightarrow \ L(w, \beta) = f(w) + \sum_{i=1}^{l} \beta_i h_i(w)$$

**存在不等式约束的函数极值问题** $\min_w \ f(w)$

$$s.t.\ g_i(w) \leq 0, i = 1, ..., k.$$
$$h_i(w) = 0, i = 1, ..., l.$$

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

**定义** $\theta_P(w) = \max_{\alpha, \beta; \alpha_i \geq 0} L(w, \alpha, \beta)$**，只有当满足约束条件，才有最大值。**

$$\Rightarrow \ \theta_P(w) = \begin{cases} f(w), \ if \ w \ satisfies \ primal \ constraints. \\ +\infty, \ otherwise \end{cases}$$

# 对偶问题

- **拉格朗日对偶性**

**原问题** $\min_w \ f(w)$ **转化为** $\min_w \theta_P(w) = \min\limits_{w} \max\limits_{\alpha,\beta;\alpha_i \geq 0} L(w,\alpha,\beta)$，**记为** $p^*$

**直接求解不容易进而转向另一个问题** $\theta_D(\alpha,\beta) = \min\limits_{w} L(w,\alpha,\beta)$，**先固定** ***α, β***，**求拉格朗日函数关于*w*的最小值，之后再求** $\theta_D(\alpha,\beta)$ **的最大值，即**

$\max\limits_{\alpha,\beta;\alpha_i \geq 0} \theta_D(\alpha,\beta) = \max\limits_{\alpha,\beta;\alpha_i \geq 0} \min\limits_{w} L(w,\alpha,\beta)$ **原问题的对偶问题，记为** $d^*$

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

**假设*f*和*g*都是凸函数，*h*是仿射的(Affine，线性函数的一般形式)，且对于所有的*i*，** $g_i(w) < 0$，**那么一定存在*w**, *α**, *β**，使*w**是原问题的解，*α**, *β**是对偶问题的解，即** $d^* = p^* = L(w^*,\alpha^*,\beta^*)$，**且*w**, *α**, *β**满足KKT条件。**

# KKT条件

- **KKT(Karush-Kuhn-Tucker)条件**

$$\frac{\partial}{\partial w_i} L(w^*, \alpha^*, \beta^*) = 0, \ \ i = 1, ...n$$

$$\frac{\partial}{\partial \beta_i} L(w^*, \alpha^*, \beta^*) = 0, \ \ i = 1, ...l$$

$$\alpha_i^* g_i(w^*) = 0, \ \ i = 1, ...k$$

**KKT对偶互补条件**

$$g_i(w^*) \leq 0, \ \ i = 1, ...k$$

$$\alpha^* \geq 0, \ \ i = 1, ...k$$

$$d^* = \min_w L(w, \alpha^*, \beta^*) \leq L(w^*, \alpha^*, \beta^*) = f(w^*) + \sum_i \alpha_i^* g_i(w^*) + \sum_i \beta_i^* h_i(w^*)$$

$$p^* = f(w^*) \Rightarrow \sum_i \alpha_i^* g_i(w^*) = 0$$

**如果*w**, *α**, *β**满足KKT条件，那么它们就是原问题和对偶问题的解。**

**补充条件隐含如果 $\alpha^* > 0$，那么$g_i(w^*) = 0$，即*w*处于可行域的边界上，是起作用的(Active)约束，而位于可行域内部的点都是不起作用的约束，其 $\alpha^* = 0$。**

# 支持向量机

- **线性支持向量机**

**KKT条件：**

$$a_n \geq 0$$

$$t_n y(x_n) - 1 \geq 0$$

$$a_n\{t_n y(x_n) - 1\} = 0$$

**支持向量：**

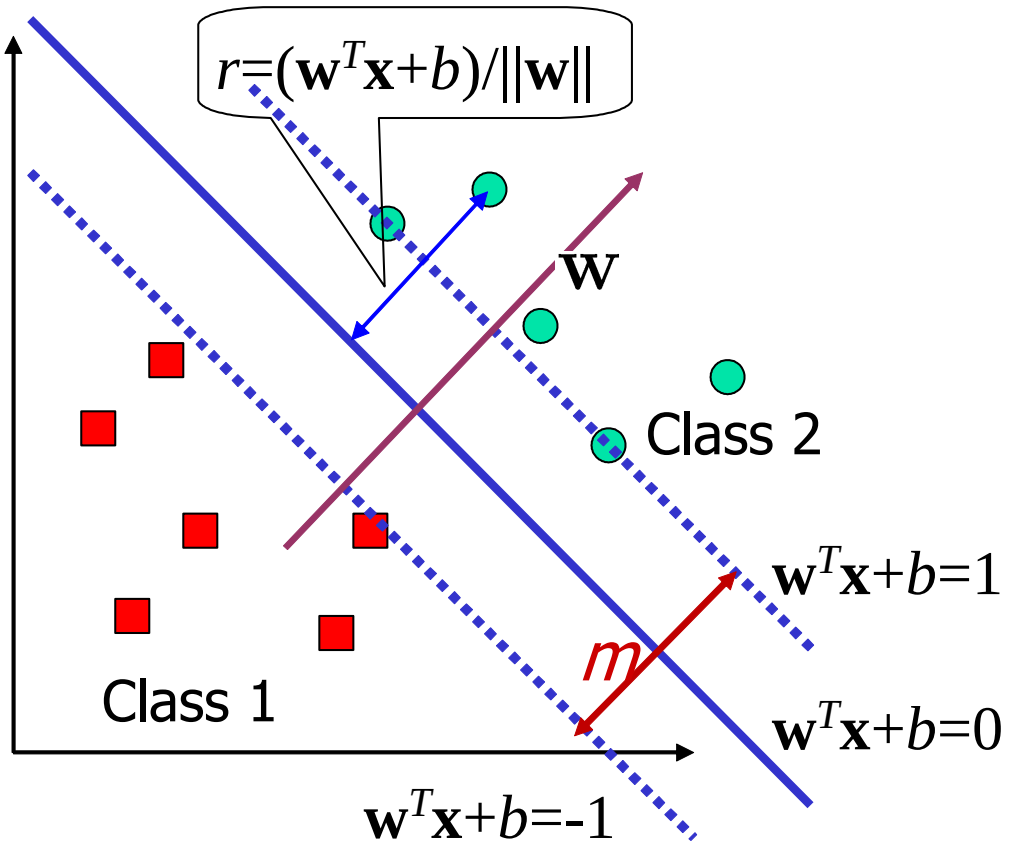
$$t_n(w^T x + b) = 1, a_n > 0$$

**非支持向量：**

$$t_n(w^T x + b) > 1, a_n = 0$$

$$y(x) = \sum_{n=1}^{N} a_n t_n x_n^T x + b$$

$$b = \frac{1}{N_S} \sum_{n \in S} (t_n - \sum_{m \in S} a_m t_m x_n^T x_m)$$




**超平面法向量是支持向量的线性组合**

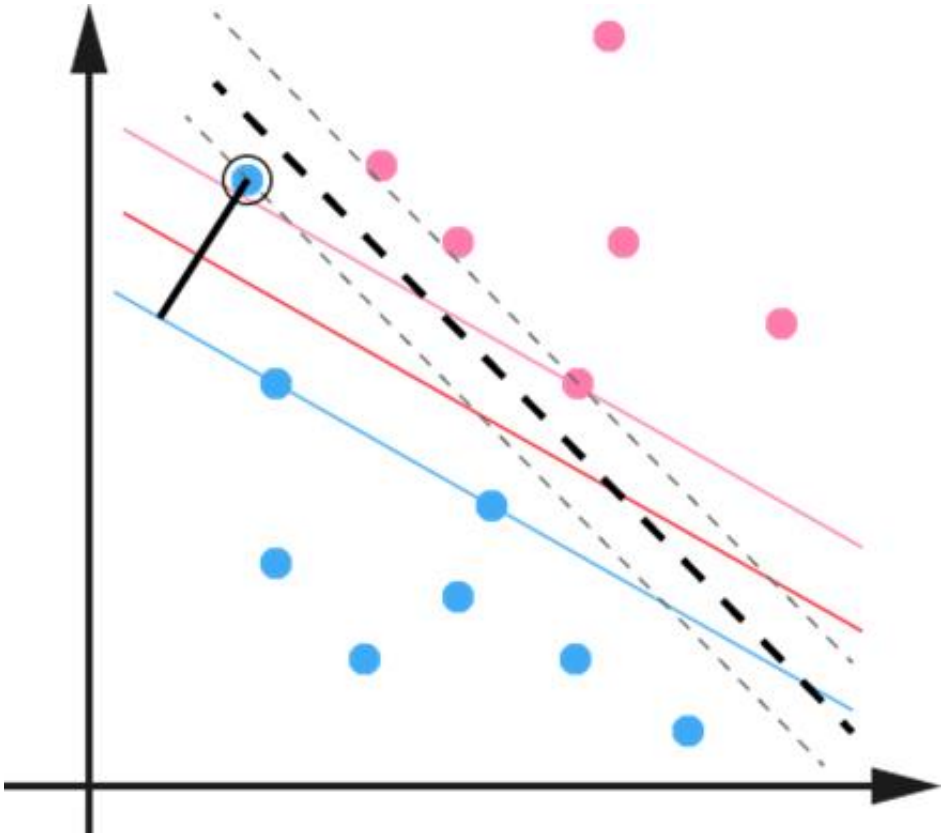
# 支持向量机

- 处理噪声和离群点

求解最优分类面的时间代价大还可能导致泛化性能差。因此，对于分布有交集的数据需要有一定范围内的“错分”，又有较大分界区域的广义最优分类面。

准确性

泛化性



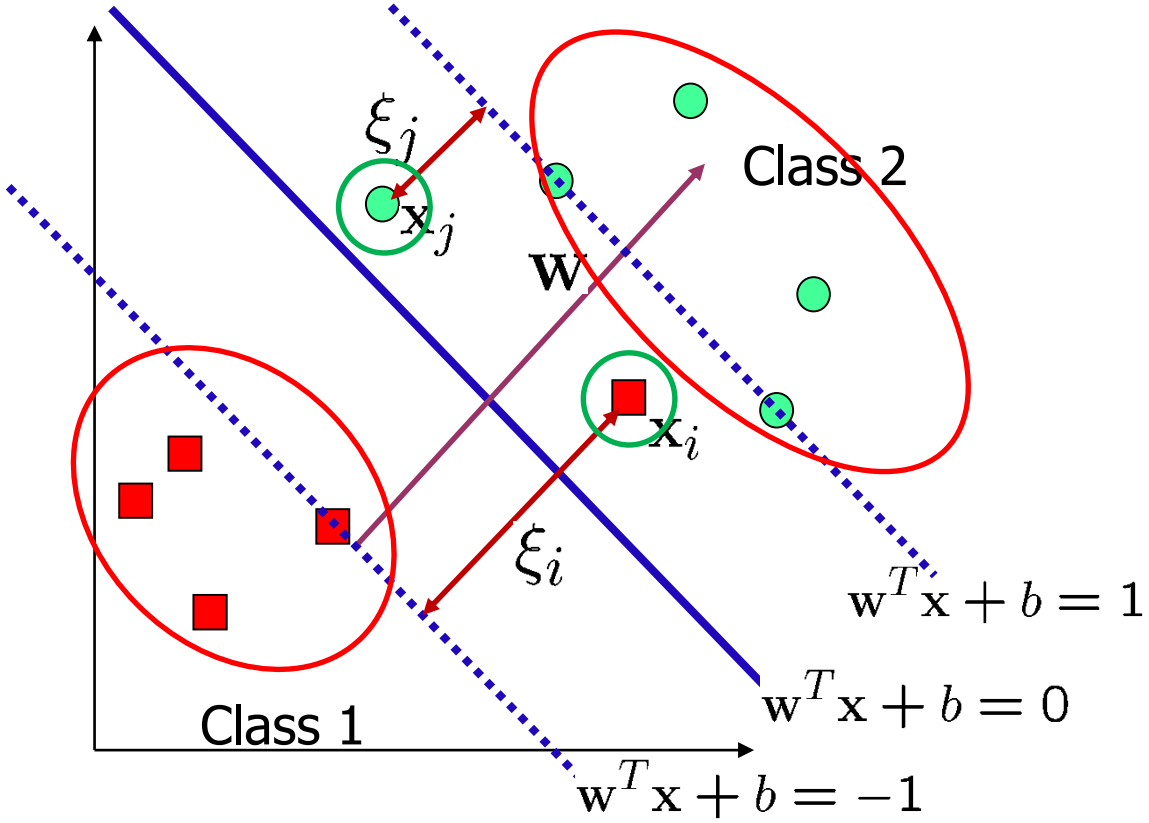
# 支持向量机

- **处理噪声和离群点**

**引入松弛变量** $\xi_n \geq 0$

$$\xi_n = 0$$
$$\xi_n = |t_n - y(x_n)|$$

$$\xi_n = 0$$
$$0 < \xi_n \leq 1$$
$$\xi_n > 1$$




**原有约束** $t_n y(x_n) \geq 1$ ➡ $t_n y(x_n) \geq 1 - \xi_n$

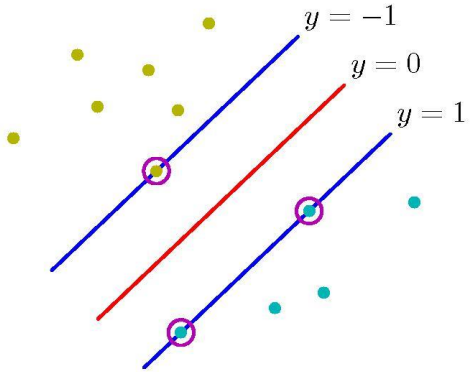
# 支持向量机

- **处理噪声和离群点**

**这种处理方式也被视为是从硬间隔(Hard Margin)向软间隔(Soft Margin)的转变。**

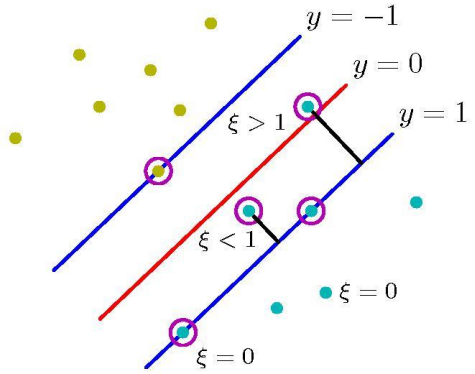
**硬间隔**

$$\min_{w,b} \frac{1}{2}\|w\|^2$$
$$s.t.\ t_n(w^T x_n + b) \geq 1,\ n = 1, ..., N$$




**软间隔**

$$\min_{w,b} \frac{1}{2}\|w\|^2 + C\sum_{n=1}^{N} \xi_n$$
$$s.t.\ t_n(w^T x_n + b) \geq 1 - \xi_n,\ n = 1, ..., N$$
$$\xi_n \geq 0$$



# 支持向量机

- **处理噪声和离群点**

**利用拉格朗日乘子法求解：**

$$L(w,b,\xi,a,\mu)=\tfrac{1}{2}\|w\|^2+C\sum_{n=1}^{N}\xi_n-\sum_{n=1}^{N}a_n\{t_n y(x_n)-1+\xi_n\}-\sum_{n=1}^{N}\mu_n\xi_n$$

$$a_n\geq 0;\mu_n\geq 0$$

**KKT条件：**

$$a_n\geq 0$$

$$t_n y(x_n)-1+\xi_n\geq 0$$

$$a_n(t_n y(x_n)-1+\xi_n)=0$$

$$\mu_n\geq 0$$

$$\xi_n\geq 0$$

$$\mu_n\xi_n=0$$

**优化*w*, *b*, $\{\xi_n\}$**

$$\frac{\partial L}{\partial w}=0\Rightarrow w=\sum_{n=1}^{N}a_n t_n x_n$$

$$\frac{\partial L}{\partial b}=0\Rightarrow\sum_{n=1}^{N}a_n t_n=0$$

$$\frac{\partial L}{\partial \xi_n}=0\Rightarrow a_n=C-\mu_n$$

**代入*L*化简：**

$$\tilde{L}(a)=\sum_{n=1}^{N}a_n-\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}a_n a_m t_n t_m x_n^T x_m$$

# 支持向量机

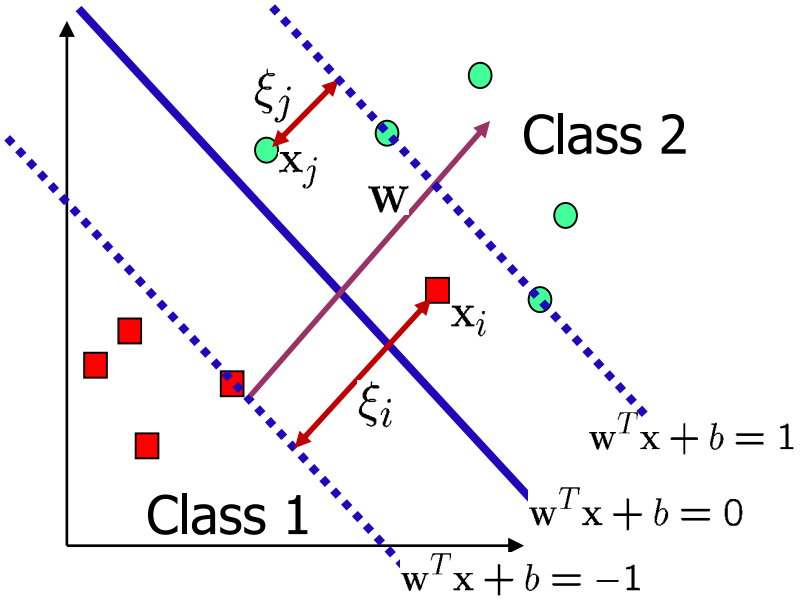
- **处理噪声和离群点**

**得到其对偶形式：**

$$\max_a \sum_{n=1}^{N} a_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} a_n a_m t_n t_m x_n^T x_m$$

$$s.t.\ 0 \le a_n \le C, n = 1, 2, ..., N$$

$$\sum_{n=1}^{N} a_n t_n = 0$$ **二次规划问题**




**对于新样本预测的分类器：**

$$y(x) = \sum_{n=1}^{N} a_n t_n x_n^T x + b$$

$$b = \frac{1}{N_{\mathcal{M}}} \sum_{n\in\mathcal{M}} (t_n - \sum_{m\in S} a_m t_m x_n^T x_m)$$

**非支持向量：** $a_n = 0$

**支持向量：** $t_n y(x_n) = 1 - \xi_n$

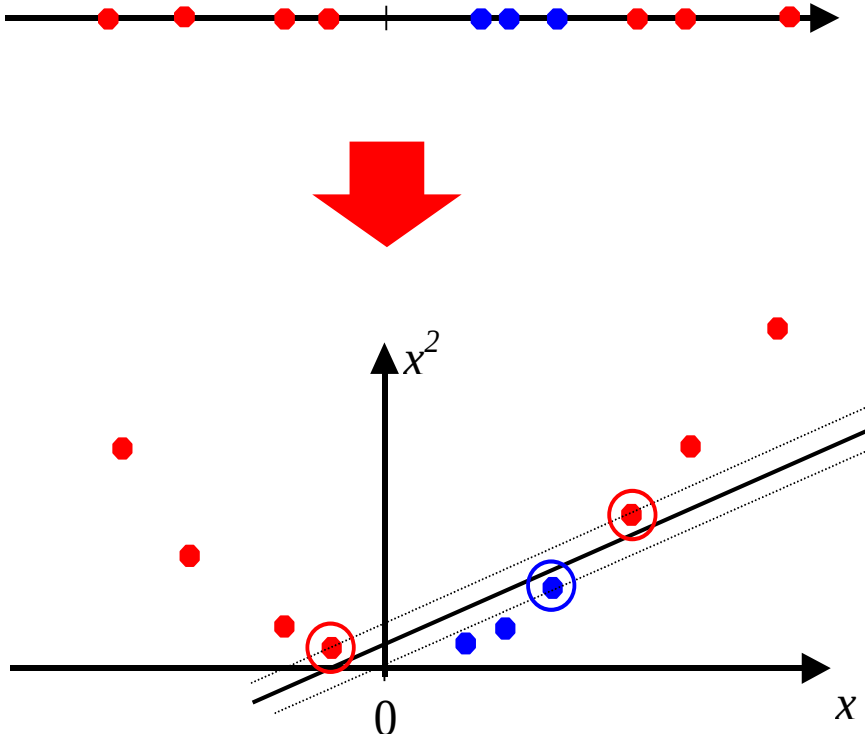
$$a_n < C \Rightarrow \mu > 0 \Rightarrow \xi_n = 0$$

$$a_n = C \Rightarrow \xi_n \le 1 \ or \ \xi_n > 1$$

# 支持向量机

- 非线性支持向量机

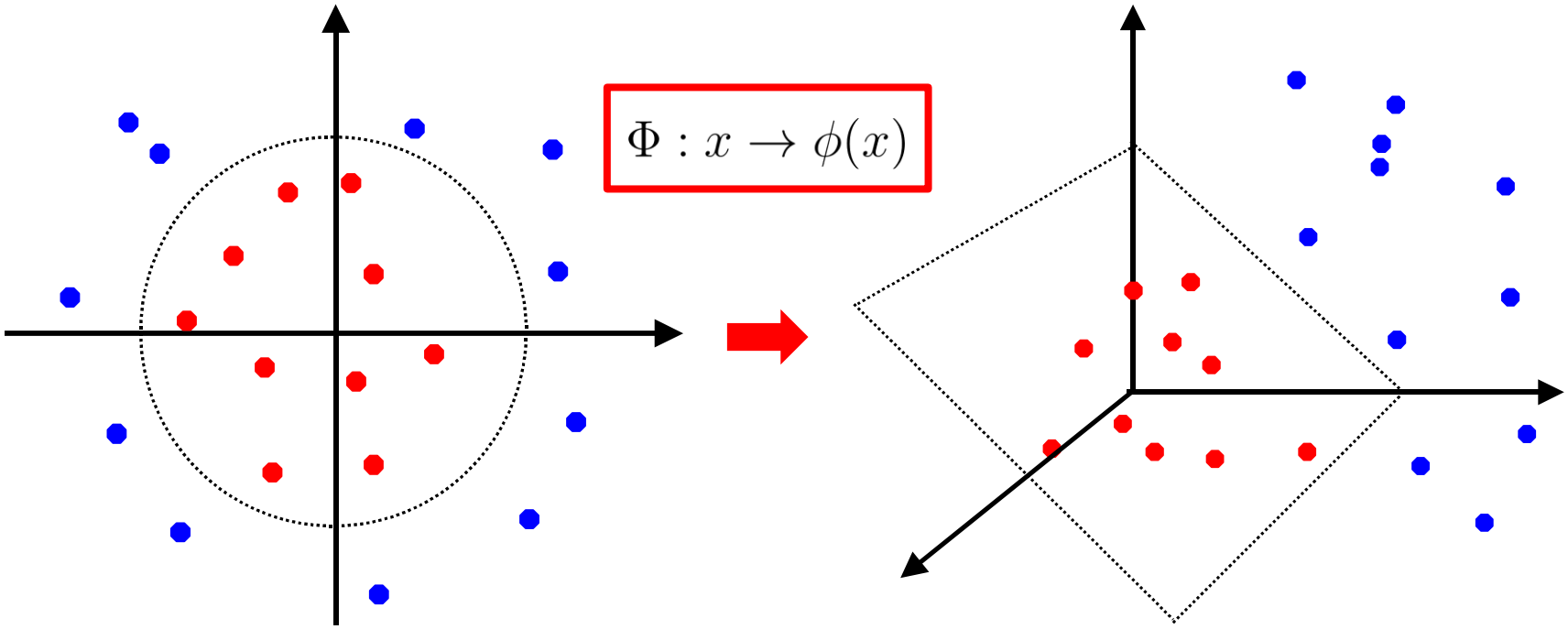
线性模型在解决复杂分类问题时适应性较差。而对于非线性可分的数据样本，可能通过适当的函数变换，将其在高维空间中转化为线性可分。



# 支持向量机

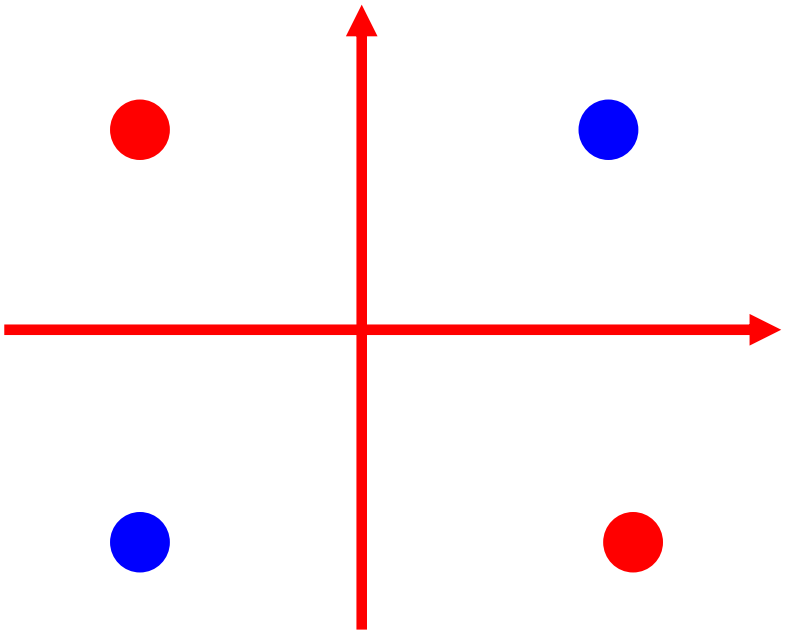
- **非线性支持向量机**

**可以把样本*x*映射到某个高维特征空间$\phi(x)$，并在其中使用线性分类器。**



# 支持向量机

- **XOR问题**



**二维样本集** $x = (x_1, x_2)$

**第一类(0, 0)和 (1, 1)，第二类(1, 0)和 (0, 1)**

**将二维数据映射到三维**

**映射函数** $\phi(x) = (x_1, x_2, x_1 x_2)$

$$(0,0) \mapsto (0,0,0)$$
$$(1,1) \mapsto (1,1,1)$$
$$(0,1) \mapsto (0,1,0)$$
$$(1,0) \mapsto (1,0,0)$$

**线性不可分 线性可分**

# 支持向量机

- 非线性支持向量机

利用一个固定的非线性映射将数据映射到特征空间学习的线性分类器等价于基于原始数据学习的非线性分类器。

$$y(x) = w^T x + b \quad \Rightarrow \quad y(x) = w^T \phi(x) + b$$

决策时

$$y(x) = \sum_{n=1}^{N} a_n t_n x_n^T x + b \quad \Rightarrow \quad y(x) = \sum_{n=1}^{N} a_n t_n k(x, x_n) + b$$

$$k(x, x_n) = \phi(x_n)^T \phi(x)$$

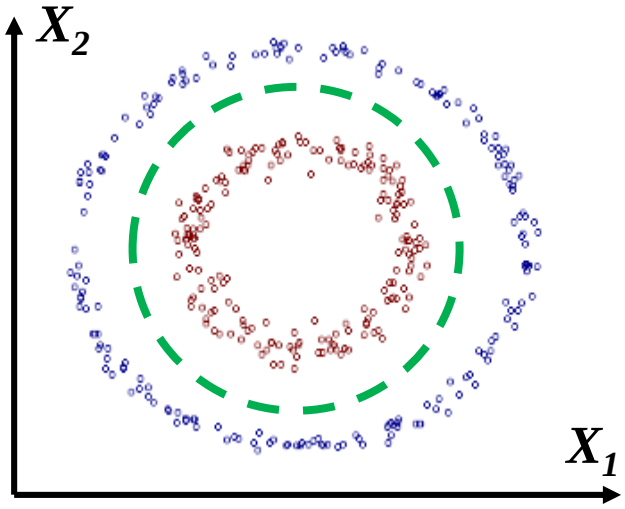
核函数

核函数在特征空间中直接计算数据映射后的内积就像在原始输入数据的函数中计算一样，大大简化了计算过程。

# 支持向量机

- **非线性支持向量机**

**利用一个固定的非线性映射将数据映射到特征空间学习的线性分类器等价于基于原始数据学习的非线性分类器。**




$$a_1 X_1 + a_2 X_1^2 + a_3 X_2 + a_4 X_2^2 + a_5 X_1 X_2 + a_6 = 0$$

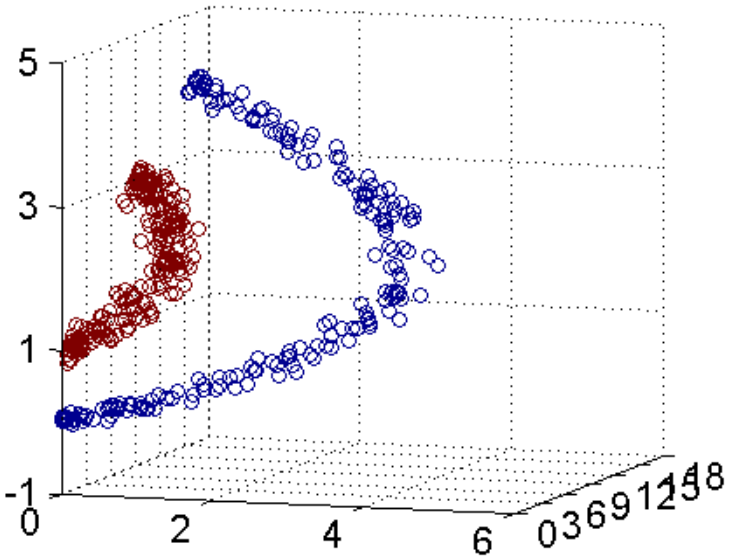
$$Z_1 = X_1; Z_2 = X_1^2; Z_3 = X_2; Z_4 = X_2^2; Z_5 = X_1 X_2$$

$$\sum_{i=1}^{5} a_i Z_i + a_6 = 0 \qquad \phi : \mathbb{R}^2 \to \mathbb{R}^5$$

$$a_1 X_1^2 + a_2 (X_2 - c)^2 + a_3 = 0$$

$$Z_1 = X_1^2; Z_2 = X_2^2; Z_3 = X_2$$

$$\phi : \mathbb{R}^2 \to \mathbb{R}^3$$



# 支持向量机

- **非线性支持向量机**

**利用一个固定的非线性映射将数据映射到特征空间学习的线性分类器等价于基于原始数据学习的非线性分类器。**

$$a_1X_1 + a_2X_1^2 + a_3X_2 + a_4X_2^2 + a_5X_1X_2 + a_6 = 0 \quad \phi : \mathbb{R}^2 \to \mathbb{R}^5$$

**原始样本增加到三维**

$$a_1X_1^3 + a_2X_2^3 + a_3X_3^3 + a_4X_1^2X_2 + a_5X_1^2X_3 + a_6X_2^2X_1 + a_7X_2^2X_3 + a_8X_3^2X_1 + a_9X_3^2X_2 + a_{10}X_1X_2X_3 + a_{11}X_1^2 + a_{12}X_2^2 + a_{13}X_3^2 + a_{14}X_1X_2 + a_{15}X_2X_3 + a_{16}X_1X_3 + a_{17}X_1 + a_{18}X_2 + a_{19}X_3 + a_{20} = 0$$

$$\phi : \mathbb{R}^3 \to \mathbb{R}^{19}$$

**维数大大增加**
**计算变得非常困难**

$$k(x_1, x_2) = (< x_1, x_2 > +1)^2$$

**利用核函数直接在原来的低维空间中进行计算不需要显式地写出映射后的结果，避免了先映射到高维空间中然后再根据内积的公式进行计算**

# 支持向量机

- **非线性支持向量机**

**根据问题和数据的不同，选择带有不同的核函数。**

**一些常用的核函数:**

**线性核:** $k(x_1, x_2) = x_1^T x_2$

**多项式核:** $k(x_1, x_2) = (< x_1, x_2 > + R)^d$

**高斯核:** $k(x_1, x_2) = \exp\{-\frac{\|x_1 - x_2\|^2}{2\sigma^2}\}$

**Sigmoid核:** $k(x_1, x_2) = \tanh(\beta_0 x_1^T x_2 + \beta_1)$

**如何判断一个函数是否可以作为核函数?**

**Mercer定理:**
$\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$**上的映射*k*是一个有效核函数(也称Mercer核函数)当且仅当对于训练样本其相应的核函数矩阵是对称半正定的,即对于任何平方可积函数*g*(*x*)有** $\int \int k(x, y)g(x)g(y)dxdy \geq 0$ **。**

# 序列最小优化算法

- **J. C. Platt(1999年提出)**

支持向量机的学习问题可以形式化为求解具有全局最优解的凸二次规划问题。许多方法可以用于求解这一问题，但当训练样本容量很大时，这些算法往往效率较低，以致无法使用。

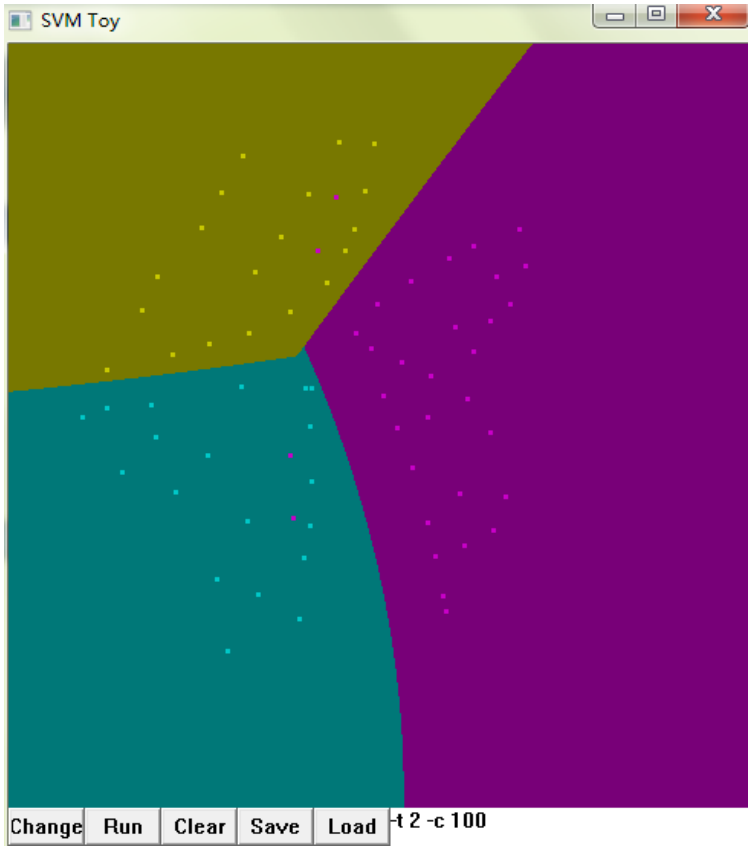
序列最小优化算法(Sequential Minimal Optimization, SMO)是一种启发式算法。基本思想是：如果所有变量都满足此优化问题的KKT条件，那么这个问题的解就得到了。

SMO算法的特点是不断地将原二次规划问题分解为只有两个变量的二次规划问题，并对子问题进行解析求解，直到所有变量都满足KKT条件为止。因为子问题解析解存在，所以每次计算子问题都很快，虽然子问题次数很多，但是总体上还是高效的。

# 支持向量机

- 支持向量机工具

**LibSVM: http://www.csie.ntu.edu.tw/~cjlin/libsvm/**



# 支持向量机应用

- 物体识别