



2020



中北大学 Android Lab
NORTH UNIVERSITY OF CHINA

机器学习培训-SVM

许超

本次培训主要目标:

- 1.理解Hard-Margin SVM算法
- 2.了解Soft-Margin SVM算法
- 3.了解核函数的作用
- 4.了解libsvm库
- 5.可以使用sklearn 对鸢尾花数据集进行分类
- 6.课下作业: 使用SVM对MNIST进行分类

Support Vector Machine

SVM有三宝：间隔，对偶，核技巧



中北大学
NORTH UNIVERSITY OF CHINA

Android Lab

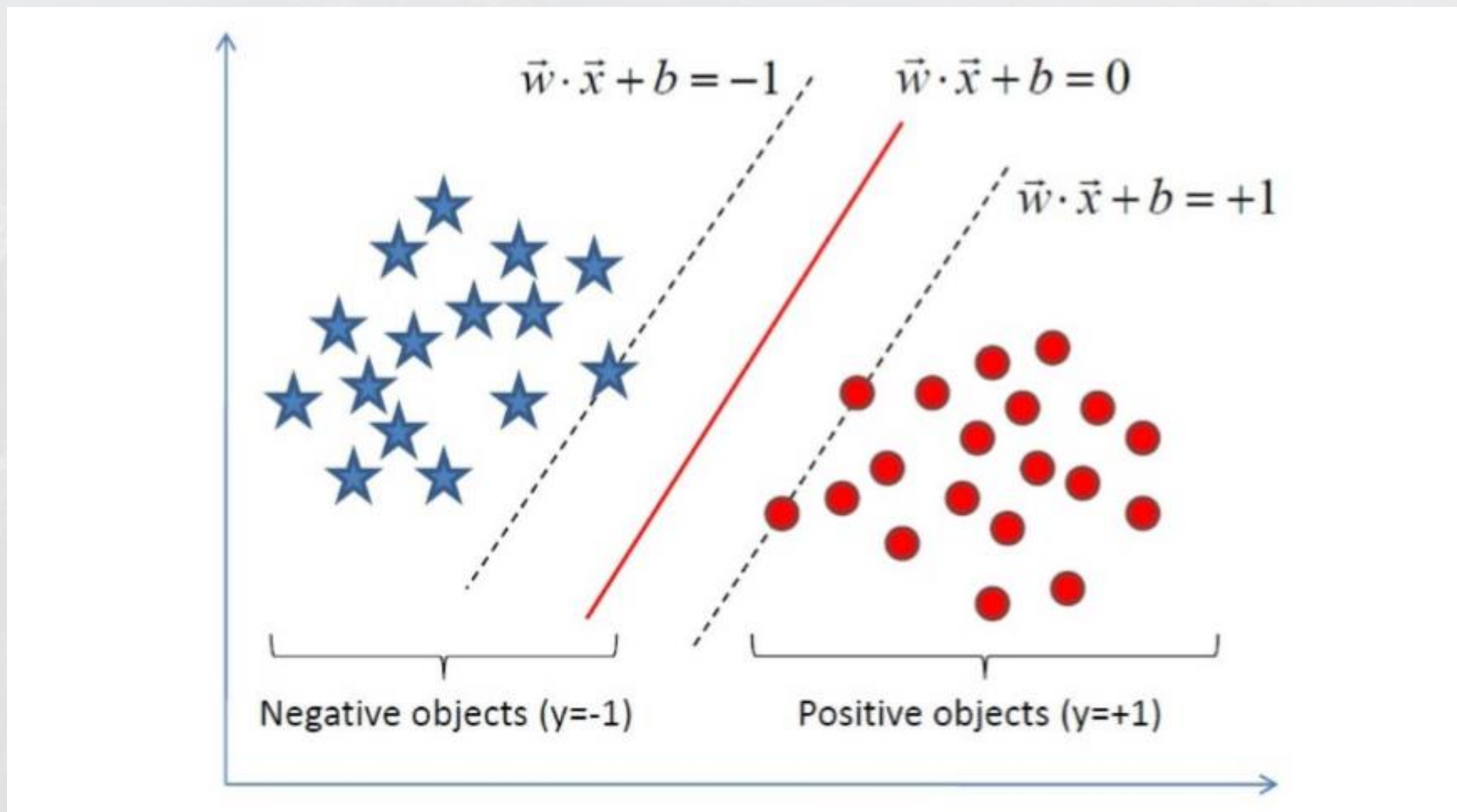
SVM

Hard-Margin SVM算法（线性）*
不允许犯错

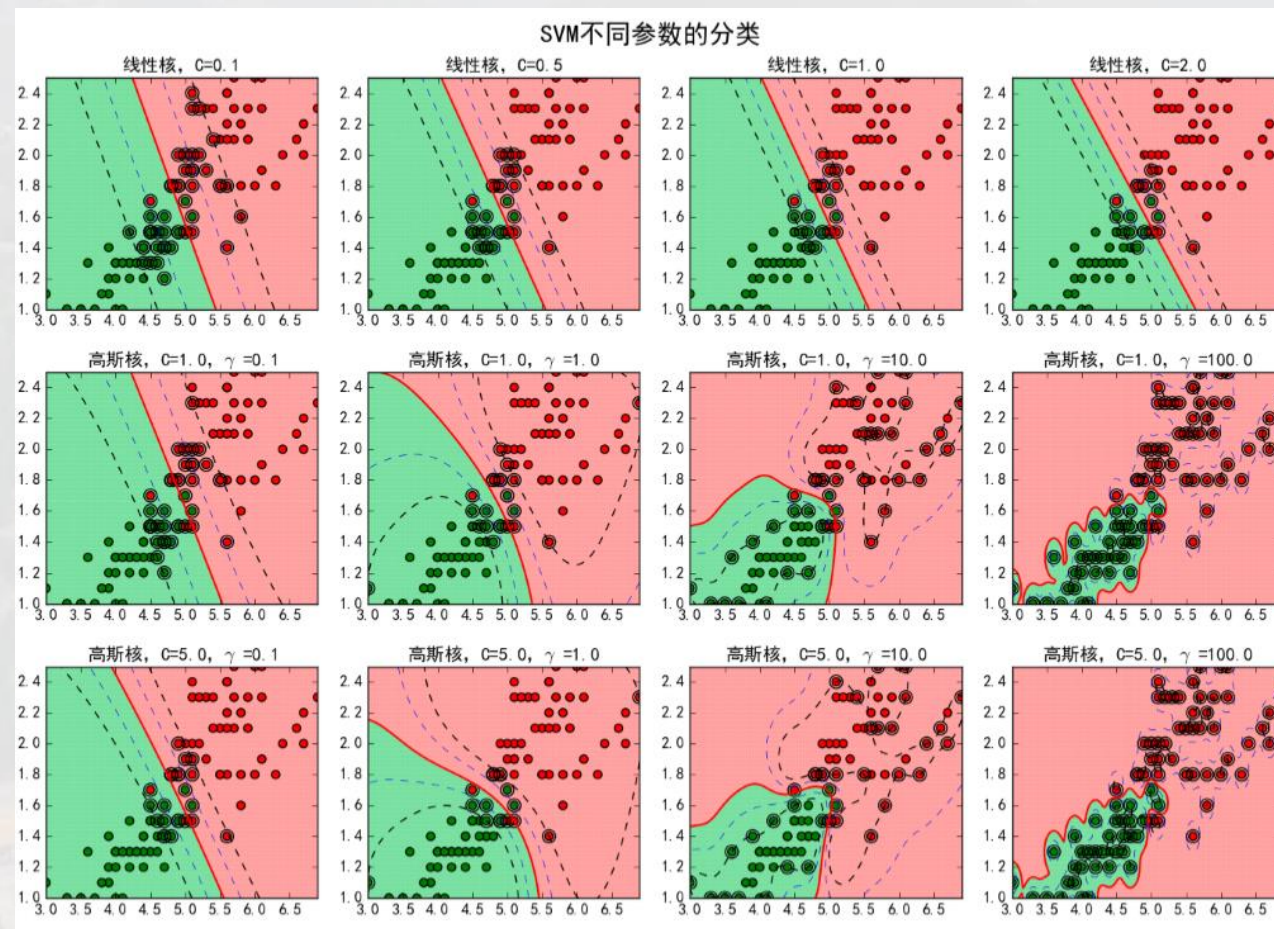
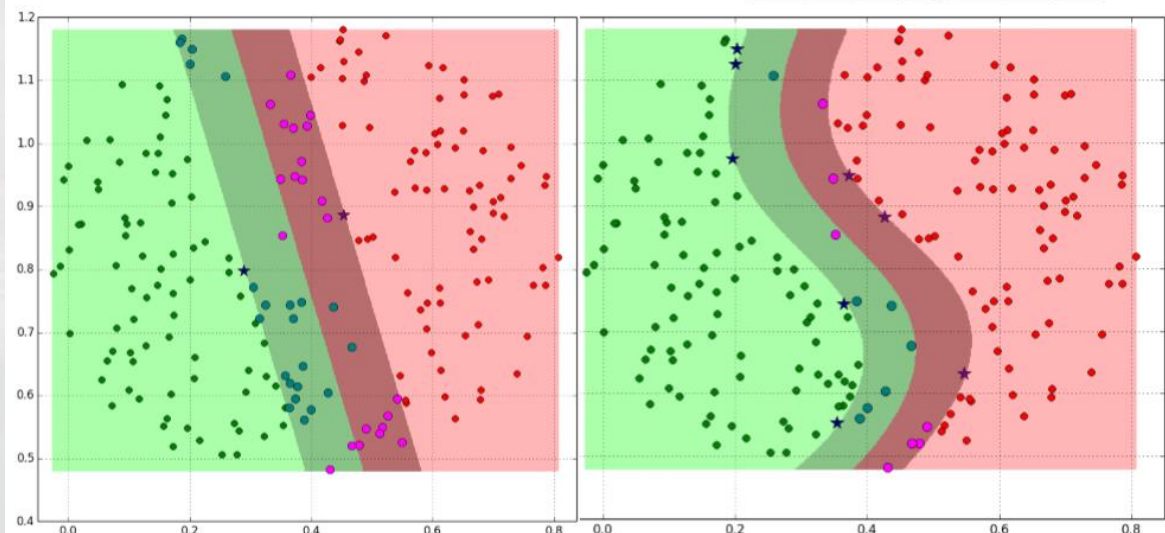
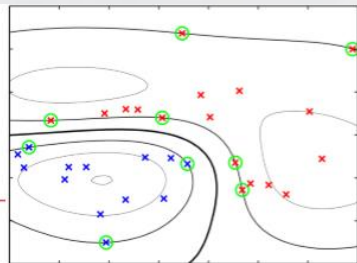
Soft-Margin SVM算法（线性）*
允许犯错

Kernel SVM算法（非线性）
允许犯错
加入核函数

线性可分支持向量机



使用核解决线性不可分



输入数据

- 假设给定一个特征空间上的训练数据集
 $T = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_N, y_N)\}$
 - 其中, $\mathbf{x}_i \in \mathbb{R}^n$, $y_i \in \{+1, -1\}$, $i=1, 2, \dots, N$ 。
- \mathbf{x}_i 为第*i*个实例(若 $n>1$, \mathbf{x}_i 为向量);
- y_i 为 \mathbf{x}_i 的类标记;
 - 当 $y_i=+1$ 时, 称 \mathbf{x}_i 为正例;
 - 当 $y_i=-1$ 时, 称 \mathbf{x}_i 为负例;
- (\mathbf{x}_i, y_i) 称为样本点。

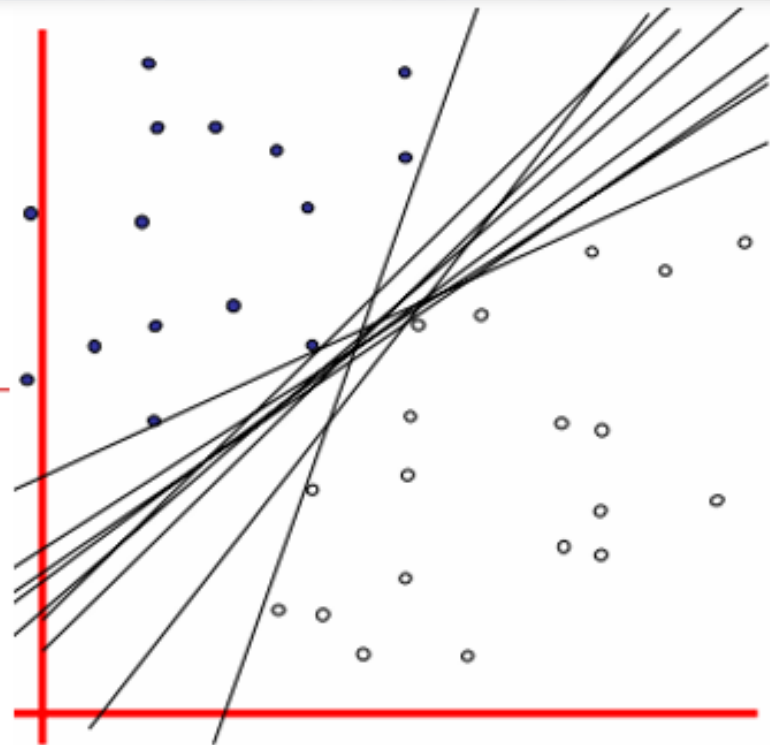
线性可分支持向量机

- 给定线性可分训练数据集，通过
间隔最大化得到的分离超平面为

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

相应的分类决策函数 $f(x) = \text{sign}(w^T \Phi(x) + b)$

该决策函数称为线性可分支持向量机。



整理符号

- 分割平面: $y(x) = w^T \Phi(x) + b$
- 训练集: x_1, x_2, \dots, x_n
- 目标值: $y_1, y_2, \dots, y_n, y_i \in \{-1, 1\}$
- 新数据的分类: $sign(y(x))$

推导目标函数

□ 根据题设 $y(x) = w^T \Phi(x) + b$

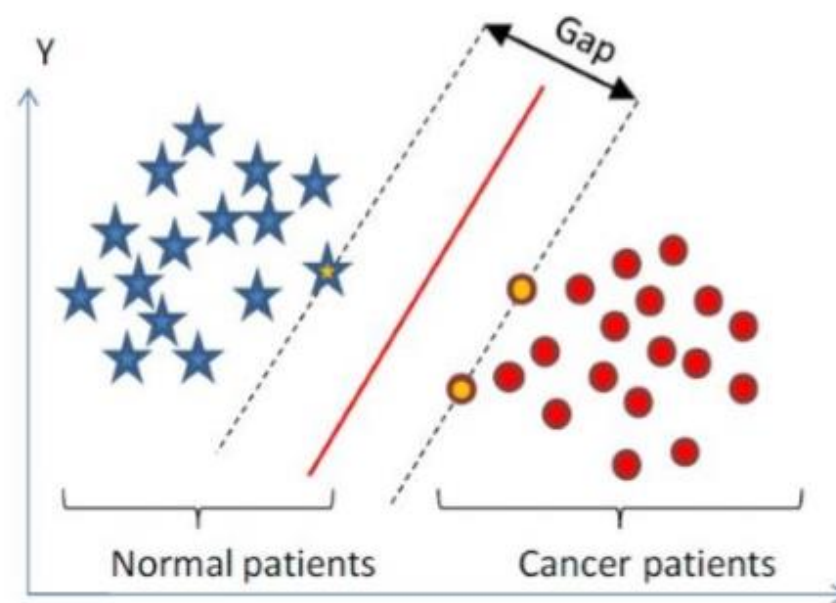
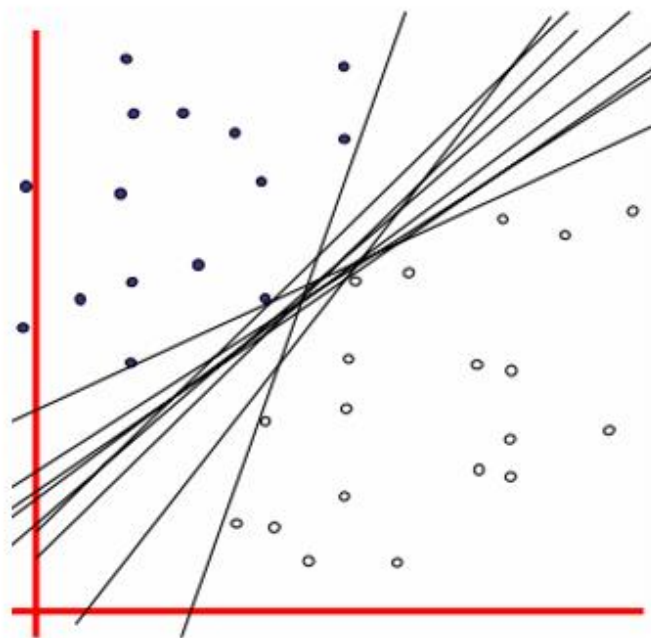
□ 有：
$$\begin{cases} y(x_i) > 0 \Leftrightarrow y_i = +1 \\ y(x_i) < 0 \Leftrightarrow y_i = -1 \end{cases} \Rightarrow y_i \cdot y(x_i) > 0$$

□ w, b 等比例缩放，则 $t \cdot y$ 的值同样缩放，从而：

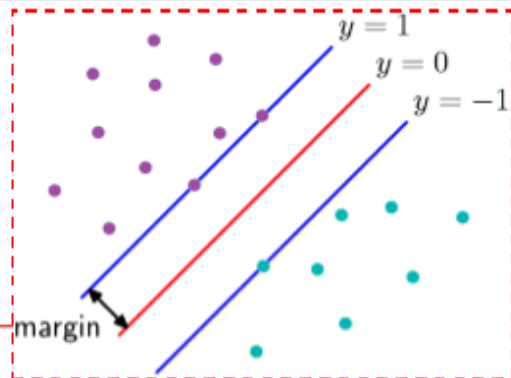
$$\frac{y_i \cdot y(x_i)}{\|w\|} = \frac{y_i \cdot (w^T \cdot \Phi(x_i) + b)}{\|w\|} \quad \text{点到直线的距离公式}$$

最大间隔分离超平面 $\frac{y_i \cdot y(x_i)}{\|w\|} = \frac{y_i \cdot (w^T \cdot \Phi(x_i) + b)}{\|w\|}$

□ 目标函数: $\arg \max_{w,b} \left\{ \frac{1}{\|w\|} \min_i [y_i \cdot (w^T \cdot \Phi(x_i) + b)] \right\}$



建立目标函数



□ 总可以通过等比例缩放 w 的方法，使得两类的函数值都满足 $|y| \geq 1$

□ 约束条件： $y_i \cdot (w^T \cdot \Phi(x_i) + b) \geq 1$

□ 原目标函数：
$$\arg \max_{w,b} \left\{ \frac{1}{\|w\|} \min_i [y_i \cdot (w^T \cdot \Phi(x_i) + b)] \right\}$$

□ 新目标函数：
$$\arg \max_{w,b} \frac{1}{\|w\|}$$

建立目标函数

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

$$s.t. \quad y_i(w^T \cdot \Phi(x_i) + b) \geq 1, \quad i = 1, 2, \dots, n$$

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

$$s.t. \quad y_i(w^T \cdot \Phi(x_i) + b) \geq 1, \quad i = 1, 2, \dots, n$$

拉格朗日乘子法 $\min_{w,b} \frac{1}{2} \|w\|^2, s.t. y_i(w^T \cdot \Phi(x_i) + b) \geq 1, i = 1, 2 \dots N$

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i (y_i (w^T \cdot \Phi(x_i) + b) - 1)$$

□ 原问题是极小极大问题

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

□ 原始问题的对偶问题，是极大极小问题

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

拉格朗日函数

- 将拉格朗日函数 $L(w,b,a)$ 分别对 w , b 求偏导并令其为0:

$$\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^n \alpha_i y_i \Phi(x_n)$$

$$\frac{\partial L}{\partial b} = 0 \Rightarrow 0 = \sum_{i=1}^n \alpha_i y_i$$

计算拉格朗日函数的对偶函数

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

$$0 = \sum_{i=1}^n \alpha_i y_i$$

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i (y_i (w^T \cdot \Phi(x_i) + b) - 1)$$

$$= \frac{1}{2} w^T w - w^T \sum_{i=1}^n \alpha_i y_i \Phi(x_i) - b \sum_{i=1}^n \alpha_i y_i + \sum_{i=1}^n \alpha_i$$

$$= \frac{1}{2} w^T \sum_{i=1}^n \alpha_i y_i \Phi(x_i) - w^T \sum_{i=1}^n \alpha_i y_i \Phi(x_i) - b \cdot 0 + \sum_{i=1}^n \alpha_i$$

$$= \sum_{i=1}^n \alpha_i - \frac{1}{2} \left(\sum_{i=1}^n \alpha_i y_i \Phi(x_i) \right)^T \sum_{i=1}^n \alpha_i y_i \Phi(x_i)$$

$$= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j \Phi^T(x_i) \Phi(x_j)$$

$$a^* = \arg \max_{\alpha} \left(\sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j \Phi^T(x_i) \Phi(x_j) \right)$$

继续求 $\min_{w,b} L(w,b,\alpha)$ 对 α 的极大

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j (\Phi(x_i) \cdot \Phi(x_j))$$

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

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

整理目标函数：添加负号

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

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

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

线性可分支持向量机学习算法

□ 构造并求解约束最优化问题

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

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

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

□ 求得最优解 α^*

线性可分支持向量机学习算法

□ 计算

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

$$b^* = y_i - \sum_{i=1}^N \alpha_i^* y_i (\Phi(x_i) \cdot \Phi(x_j))$$

□ 求得分离超平面

$$w^* \Phi(x) + b^* = 0$$

□ 分类决策函数

$$f(x) = \text{sign}(w^* \Phi(x) + b^*)$$

核函数

- 可以使用核函数，将原始输入空间映射到新的特征空间，从而，使得原本线性不可分的样本可能在核空间可分。
 - 多项式核函数： $\kappa(x_1, x_2) = (x_1 \cdot x_2 + c)^d$
 - 高斯核RBF函数： $\kappa(x_1, x_2) = \exp(-\gamma \cdot \|x_1 - x_2\|^2)$
 - Sigmoid核函数： $\kappa(x_1, x_2) = \tanh(x_1 \cdot x_2 + c)$
- 在实际应用中，往往依赖先验领域知识/交叉验证等方案才能选择有效的核函数。
 - 没有更多先验信息，则使用高斯核函数

多项式核函数 $\kappa(\vec{x}, \vec{y}) = (\vec{x} \cdot \vec{y})^2$

$$\kappa(\vec{x}, \vec{y}) = (\vec{x} \cdot \vec{y})^2$$

$$\Rightarrow \left(\sum_{i=1}^n x_i y_i \right)^2$$

$$= \sum_{i=1}^n \sum_{j=1}^n x_i x_j y_i y_j$$

$$= \sum_{i=1}^n \sum_{j=1}^n (x_i x_j)(y_i y_j)$$

$$\Rightarrow \Phi(\vec{x}) = \text{vec}(x_i x_j) \Big|_{i,j=1}^n$$

特殊的，若 $n=3$ ，即： $\Phi(\vec{x}) =$

$$\begin{pmatrix} x_1 x_1 \\ x_1 x_2 \\ x_1 x_3 \\ x_2 x_1 \\ x_2 x_2 \\ x_2 x_3 \\ x_3 x_1 \\ x_3 x_2 \\ x_3 x_3 \end{pmatrix}$$

多项式核 $\kappa(\vec{x}, \vec{y}) = (\vec{x} \cdot \vec{y} + c)^2$

$$\kappa(\vec{x}, \vec{y}) = (\vec{x} \cdot \vec{y} + c)^2$$

$$\Rightarrow (\vec{x} \cdot \vec{y})^2 + 2c\vec{x} \cdot \vec{y} + c^2$$

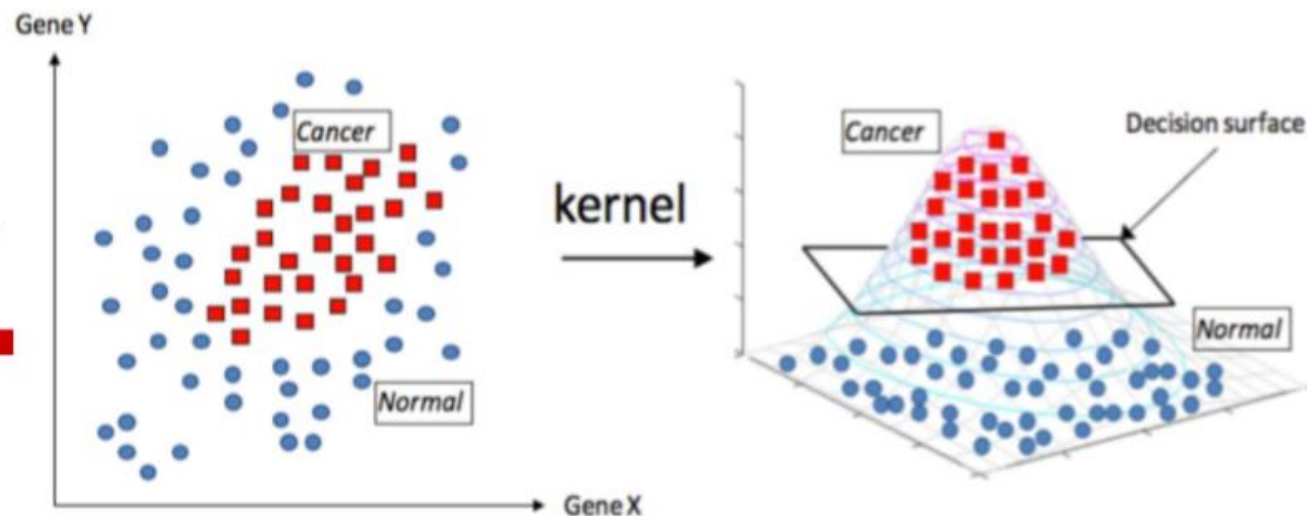
$$= \sum_{i=1}^n \sum_{j=1}^n (x_i x_j)(y_i y_j) + \sum_{i=1}^n (\sqrt{2c} x_i \cdot \sqrt{2c} x_j) + c^2$$

$$\Rightarrow \Phi(\vec{x}) = \left(\text{vec}(x_i x_j) \Big|_{i,j=1}^n, \text{vec}(\sqrt{2c} x_i) \Big|_{i=1}^n, c \right)$$

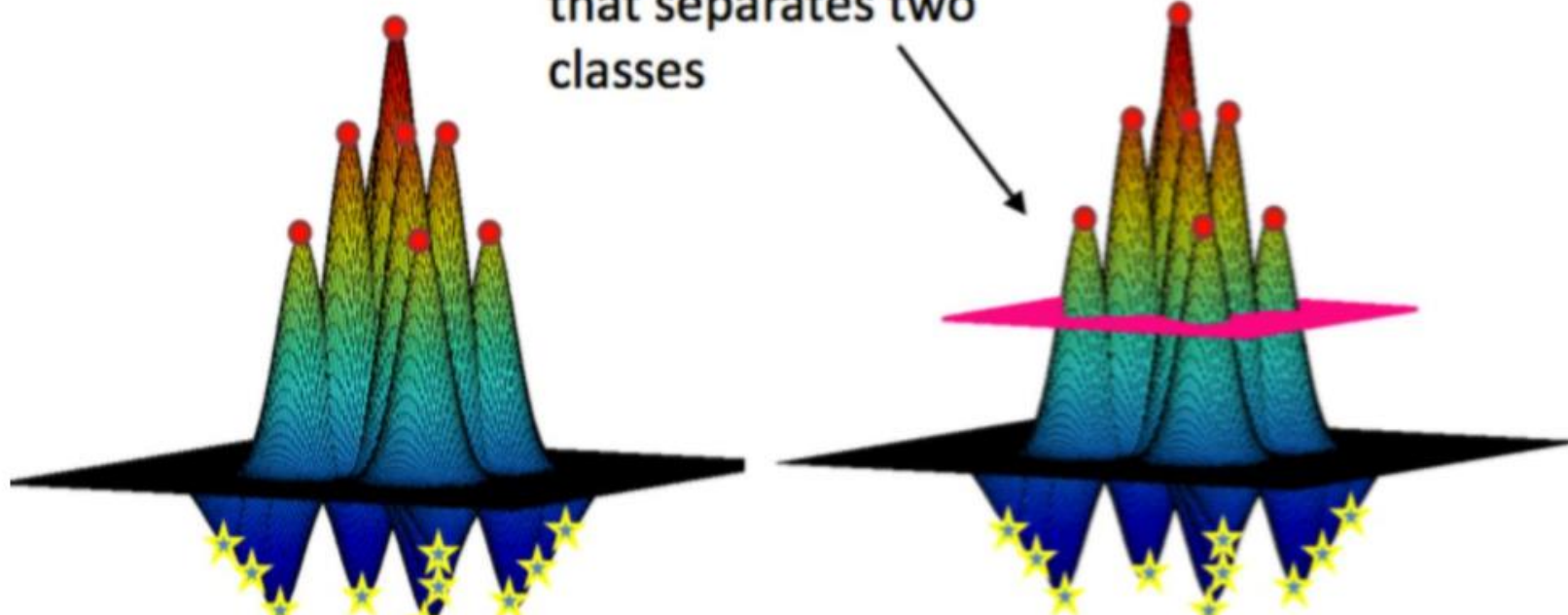
特殊的，若 $n=3$ ，即： $\Phi(\vec{x}) =$

$$\begin{pmatrix} x_1 x_1 \\ x_1 x_2 \\ x_1 x_3 \\ x_2 x_1 \\ x_2 x_2 \\ x_2 x_3 \\ x_3 x_1 \\ x_3 x_2 \\ x_3 x_3 \\ \sqrt{2c} x_1 \\ \sqrt{2c} x_2 \\ \sqrt{2c} x_3 \\ c \end{pmatrix}$$

核函数映射

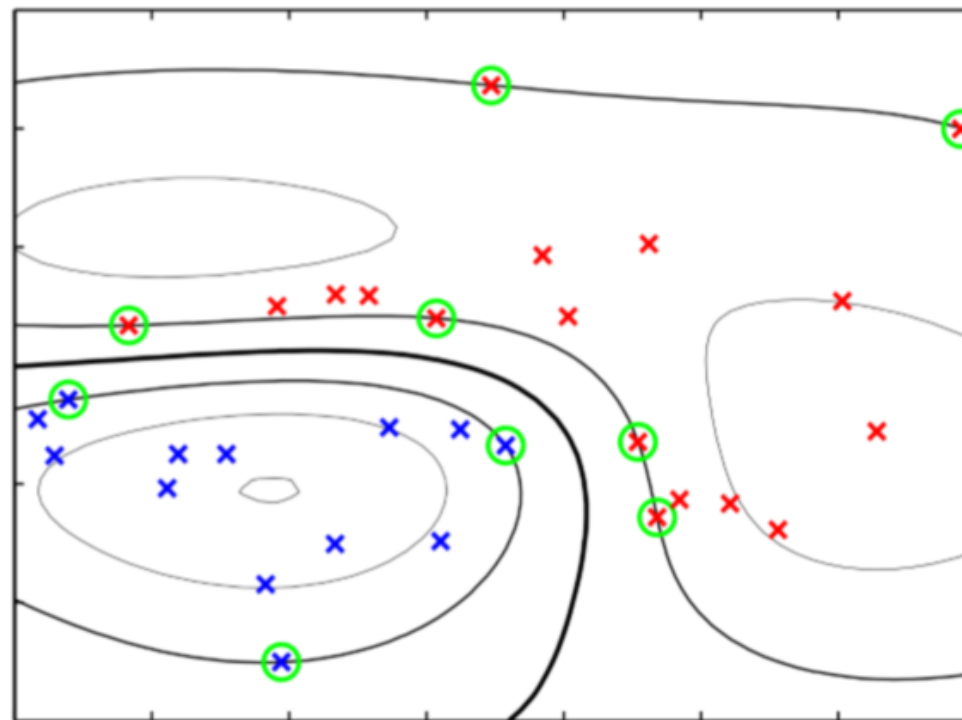


Linear hyperplane
that separates two
classes



高斯核

- ❑ 粗线是分割超“平面”
- ❑ 其他线是 $y(x)$ 的等高线
- ❑ 绿色圈点是支持向量点



高斯核是无穷维的 $e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots + \frac{x^n}{n!} + R_n$

$$\begin{aligned}
 \kappa(x_1, x_2) &= e^{-\frac{\|x_1 - x_2\|^2}{2\sigma^2}} = e^{-\frac{(x_1 - x_2)^2}{2\sigma^2}} = e^{-\frac{x_1^2 + x_2^2 - 2x_1x_2}{2\sigma^2}} = e^{-\frac{x_1^2 + x_2^2}{2\sigma^2}} \cdot e^{\frac{x_1x_2}{\sigma^2}} \\
 &= e^{-\frac{x_1^2 + x_2^2}{2\sigma^2}} \cdot \left(1 + \frac{1}{\sigma^2} \cdot \frac{x_1x_2}{1!} + \left(\frac{1}{\sigma^2}\right)^2 \cdot \frac{(x_1x_2)^2}{2!} + \left(\frac{1}{\sigma^2}\right)^3 \cdot \frac{(x_1x_2)^3}{3!} + \cdots + \left(\frac{1}{\sigma^2}\right)^n \cdot \frac{(x_1x_2)^n}{n!} + \cdots \right) \\
 &= e^{-\frac{x_1^2 + x_2^2}{2\sigma^2}} \cdot \left(1 \cdot 1 + \frac{1}{1!} \frac{x_1}{\sigma} \cdot \frac{x_2}{\sigma} + \frac{1}{2!} \frac{x_1^2}{\sigma^2} \cdot \frac{x_2^2}{\sigma^2} + \frac{1}{3!} \frac{x_1^3}{\sigma^3} \cdot \frac{x_2^3}{\sigma^3} + \cdots + \frac{1}{n!} \frac{x_1^n}{\sigma^n} \cdot \frac{x_2^n}{\sigma^n} + \cdots \right) \\
 &= \Phi(x_1)^T \cdot \Phi(x_2)
 \end{aligned}$$

□ 其中 $\Phi(x) = e^{-\frac{x^2}{2\sigma^2}} \left(1, \sqrt{\frac{1}{1!}} \frac{x}{\sigma}, \sqrt{\frac{1}{2!}} \frac{x^2}{\sigma^2}, \sqrt{\frac{1}{3!}} \frac{x^3}{\sigma^3}, \cdots, \sqrt{\frac{1}{n!}} \frac{x^n}{\sigma^n}, \cdots \right)$

理论拓展：

1.SVM多分类 (比如1,2,3,4)

1-234 2-134 3-124 4-123 or
1-2 1-3 1-4 2-3 2-4 3-4

2.CNN+SVM

CNN提取特征，SVM进行分类

3.SVM是可做回归——SVR



OVER