支持向量机

七月算法 邹博

2015年4月18日

复习：对偶问题

- 一般优化问题的Lagrange乘子法

$$
\begin{aligned}
\text{minimize} \quad & f_0(x), \quad x \in \mathbf{R}^n \\
\text{subject to} \quad & f_i(x) \le 0, \quad i = 1, \dots, m \\
& h_j(x) = 0, \quad j = 1, \dots, p
\end{aligned}
$$

- Lagrange函数

$$L(x, \lambda, v) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{j=1}^{p} v_j h_j(x)$$

  - 对固定的x，Lagrange函数L(x, λ ,v)为关于λ和v的仿射函数

复习：Lagrange对偶函数(dual function)

- Lagrange对偶函数

$$g(\lambda,\nu)=\inf_{x\in D} L(x,\lambda,\nu)=\inf_{x\in D}\left(f_0(x)+\sum_{i=1}^{m}\lambda_i f_i(x)+\sum_{i=1}^{p}\nu_i h_i(x)\right)$$

- 若没有下确界，定义：

$$g(\lambda,\nu)=-\infty$$

- 根据定义，显然有：对∀ λ>0，∀v，若原优化问题有最优值p*，则

$$g(\lambda,\nu)\le p^*$$

- 进一步：Lagrange对偶函数为凹函数。

线性方程的最小二乘问题

- 原问题 $\text{minimize} \quad x^T x, \quad x \in \mathbf{R}^n$

  $\text{subject to} \quad Ax = b$
- Lagrange函数

$$L(x, \nu) = x^T x + \nu^T (Ax - b)$$

- Lagrange对偶函数

$$g(\nu) = -\frac{1}{4}\nu^T A A^T \nu - b^T \nu$$

  - 对L求x的偏导，带入L
  - 对g求v的偏导

强对偶条件

- 若要对偶函数的最大值即为原问题的最小值，考察需要满足的条件：

$$
\begin{aligned}
f_0(x^\star) &= g(\lambda^\star, \nu^\star) \\
&= \inf_x \left(f_0(x) + \sum_{i=1}^{m} \lambda_i^\star f_i(x) + \sum_{i=1}^{p} \nu_i^\star h_i(x) \right) \\
&\le f_0(x^\star) + \sum_{i=1}^{m} \lambda_i^\star f_i(x^\star) + \sum_{i=1}^{p} \nu_i^\star h_i(x^\star) \\
&\le f_0(x^\star).
\end{aligned}
$$

Karush-Kuhn-Tucker (KKT)条件

$$
\begin{aligned}
f_0(x^\star) &= g(\lambda^\star, \nu^\star) = \inf_x \left(f_0(x) + \sum_{i=1}^{m} \lambda_i^\star f_i(x) + \sum_{i=1}^{p} \nu_i^\star h_i(x) \right) \\
&\le f_0(x^\star) + \sum_{i=1}^{m} \lambda_i^\star f_i(x^\star) + \sum_{i=1}^{p} \nu_i^\star h_i(x^\star) \\
&\le f_0(x^\star)
\end{aligned}
$$

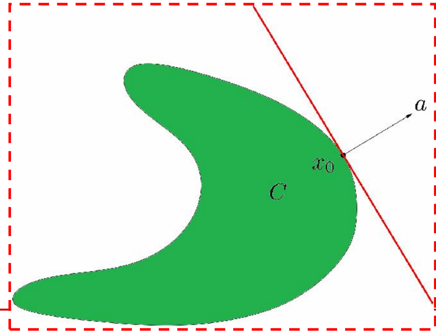
$$
\begin{aligned}
f_i(x^\star) &\le 0, \quad i = 1, \ldots, m \\
h_i(x^\star) &= 0, \quad i = 1, \ldots, p \\
\lambda_i^\star &\ge 0, \quad i = 1, \ldots, m \\
\lambda_i^\star f_i(x^\star) &= 0, \quad i = 1, \ldots, m
\end{aligned}
$$

$$
\nabla f_0(x^\star) + \sum_{i=1}^{m} \lambda_i^\star \nabla f_i(x^\star) + \sum_{i=1}^{p} \nu_i^\star \nabla h_i(x^\star) = 0
$$

主要内容和目标

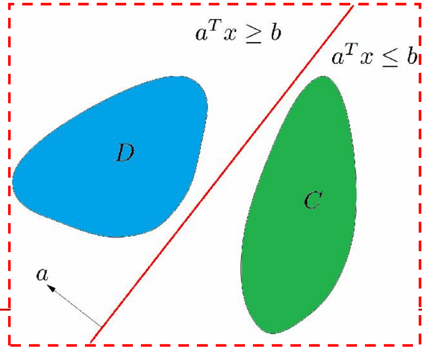
- 理解支持向量机SVM的原理和目标
- 掌握支持向量机的计算过程和算法步骤
- 对线性不可分的数据，理解软间隔最大化的含义
- 了解核函数的思想
- 了解SMO算法的过程

支撑超平面



- 设集合C，x0为C边界上的点。若存在a≠0，满足对任意x∈C，都有 $a^T x \le a^T x_0$ 成立，则称超平面 $\{x \mid a^T x = a^T x_0\}$ 为集合C在点x0处的支撑超平面。
- 凸集边界上任意一点，均存在支撑超平面。
- 反之，若一个闭的非中空(内部点不为空)集合，在边界上的任意一点存在支撑超平面，则该集合为凸集。

分割超平面



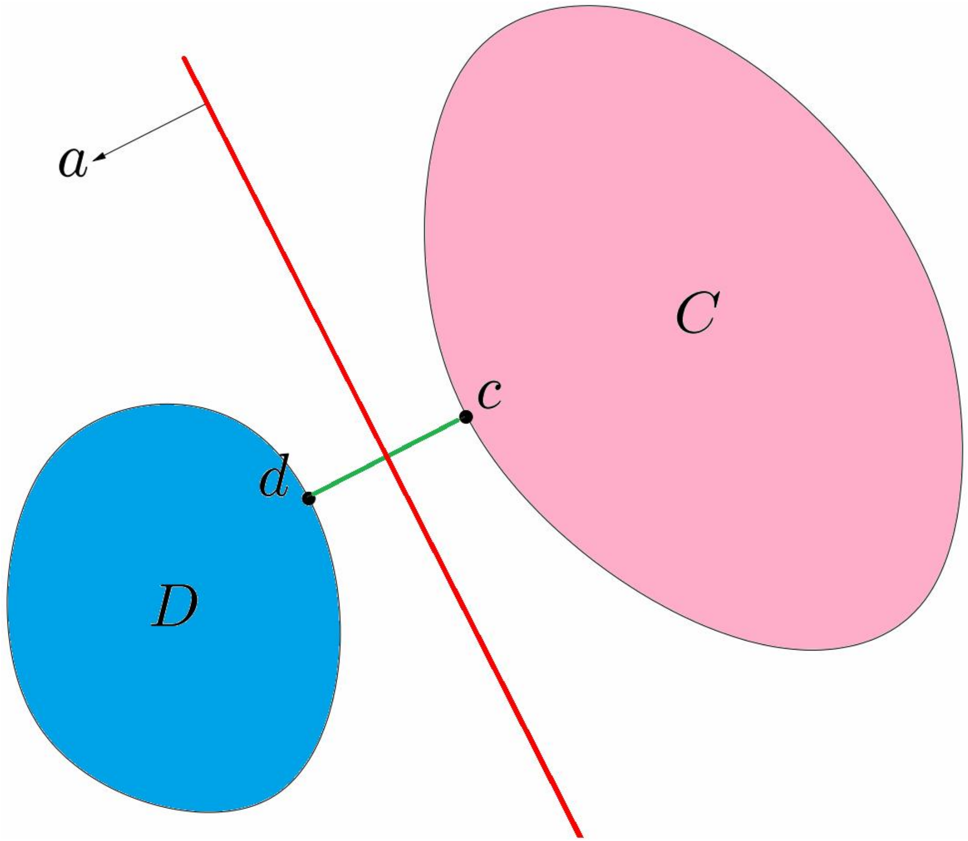
- 设C和D为两不相交的凸集，则存在超平面P，P可以将C和D分离。

$$\forall x \in C, a^T x \le b \text{ 且 } \forall x \in D, a^T x \ge b$$

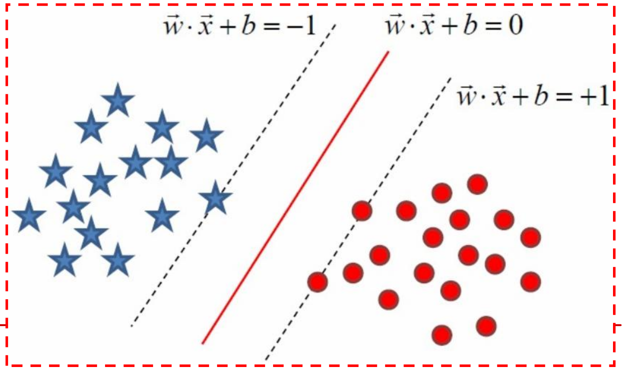
- 注意上式中可以取等号：
  - 所以：逆命题：“若两个凸集C和D的分割超平面存在，C和D不相交”为假命题。
  - 加强条件：若两个凸集至少有一个是开集，那么当且仅当存在分割超平面，它们不相交。

分割超平面的构造

- 两个集合的距离，定义为两个集合间元素的最短距离。
- 做集合C和集合D最短线段的垂直平分线。

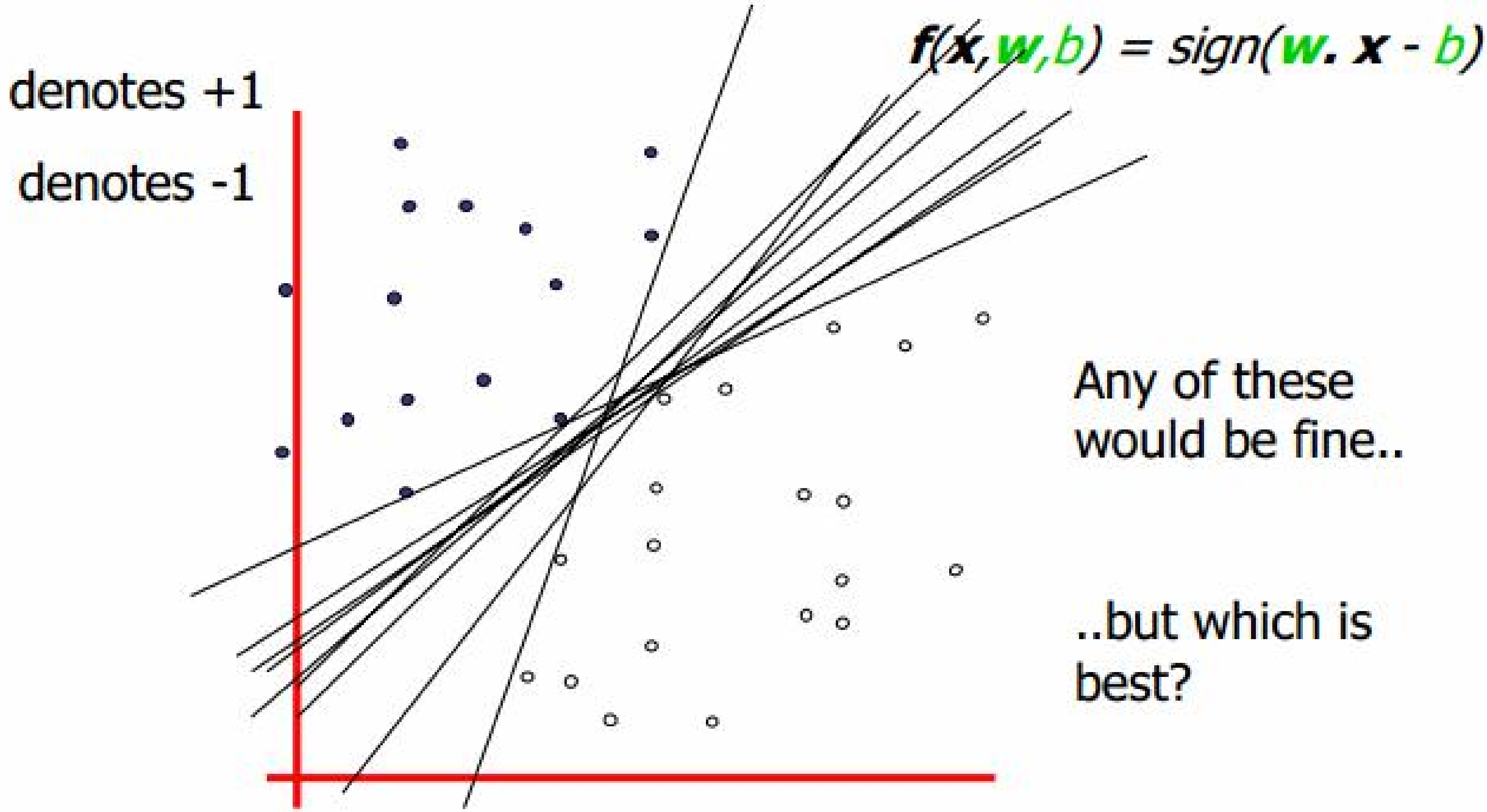


分割超平面的思考



- 如何定义两个集合的“最优”分割超平面？
  - 找到集合“边界”上的若干点，以这些点为“基础”计算超平面的方向；以两个集合边界上的这些点的平均作为超平面的“截距”
  - 支持向量：support vector
- 若两个集合有部分相交，如何定义超平面，使得两个集合“尽量”分开？
  - 注：上述“集合”不一定是凸集，可能是由若干离散点组成。若一组集合为(**x**,1)，另一组集合为(**x**,2)，则为机器学习中的分类问题。

线性分类问题



denotes +1
denotes -1
f(x,w,b) = sign(w. x - b)
Any of these would be fine..
..but which is best?

输入数据

- 假设给定一个特征空间上的训练数据集 $T=\{(\mathbf{x_1},t_1), (\mathbf{x_2},t_2)\ldots(\mathbf{x_N},t_N)\}$
  - 其中，$\mathbf{x_i}\in R^n$，$t_i\in\{+1,-1\}$，$i=1,2,\ldots N$。
- $\mathbf{x_i}$为第i个实例(若n>1， $\mathbf{x_i}$为向量)；
- t_i为$\mathbf{x_i}$的类标记；
  - 当$t_i=+1$时，称$\mathbf{x_i}$为正例；
  - 当$\mathbf{t_i}=-1$时，称xi为负例；
- $(\mathbf{x_i},t_i)$称为样本点。

各种概念

- 线性可分支持向量机
  - 硬间隔最大化hard margin maximization
  - 硬间隔支持向量机
- 线性支持向量机
  - 软间隔最大化soft margin maximization
  - 软间隔支持向量机
- 非线性支持向量机
  - 核函数kernel function

- 注：以上概念的提法，各个文献并不十分统一。

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

$$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) + b$$

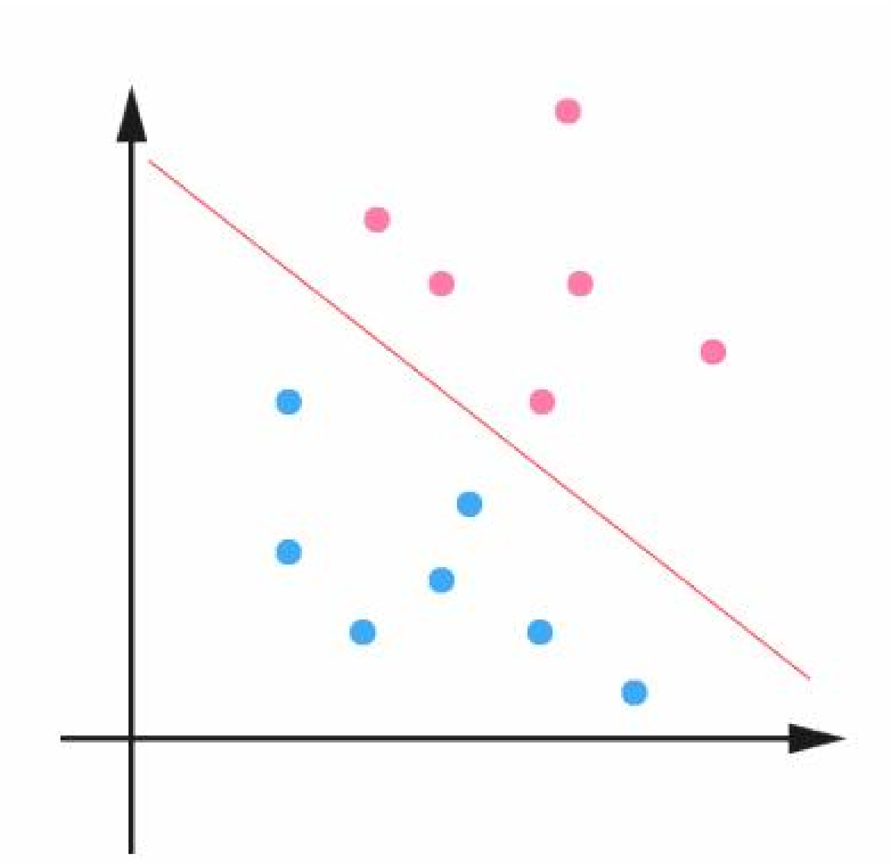
  相应的分类决策函数 $f(x)=sign\left(w^T\Phi(x)+b\right)$

  该决策函数称为线性可分支持向量机。
- $\phi(\mathrm{x})$是某个确定的特征空间转换函数，它的作用是将x映射到(更高的)维度。
  - 最简单直接的：$\phi(\mathrm{x})=\mathrm{x}$
- 稍后会看到，求解分离超平面问题可以等价为求解相应的**凸二次规划问题**。

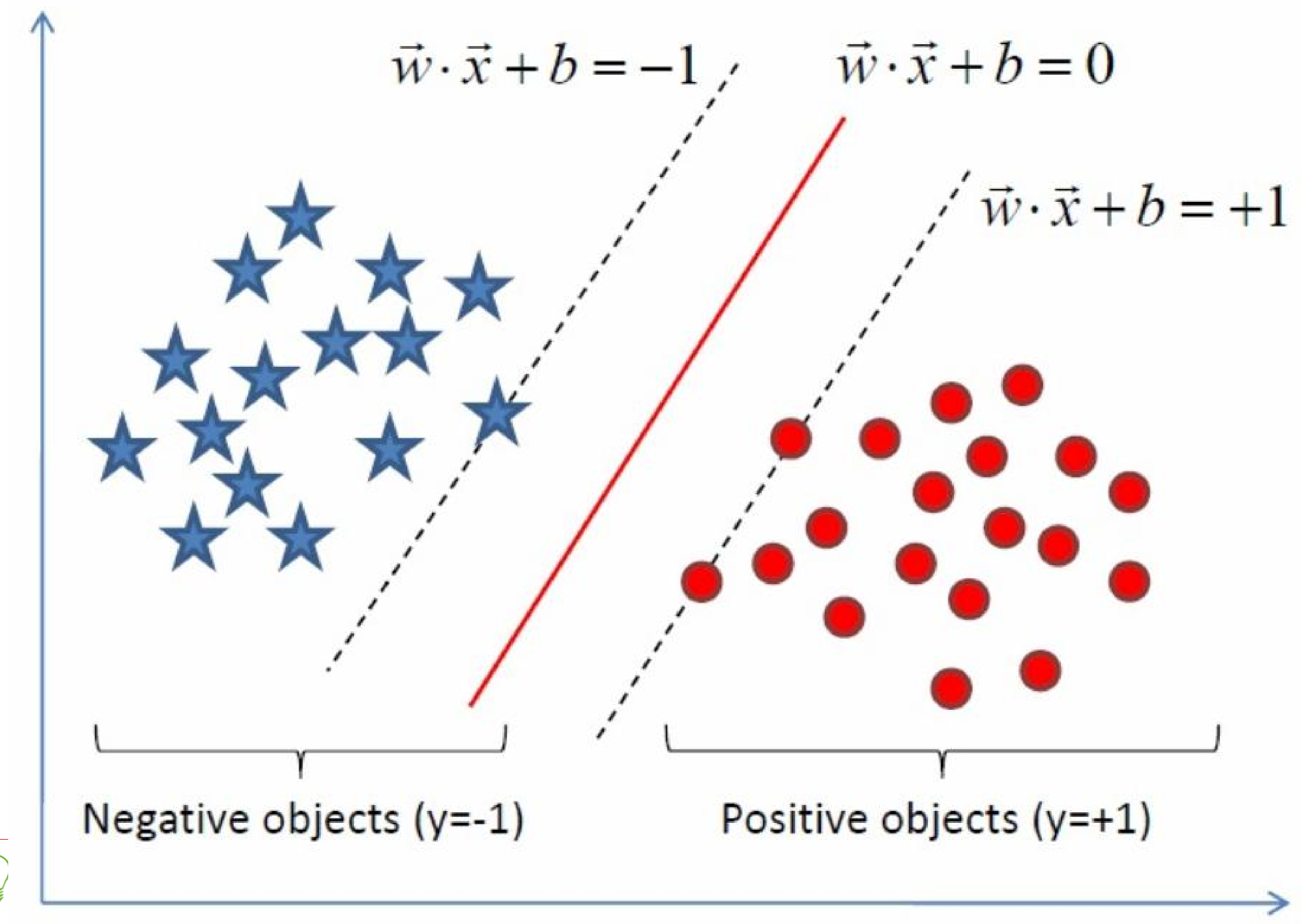
整理符号

- 分割平面：$y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) + b$
- 训练集：$\mathbf{x}_1, \ldots, \mathbf{x}_N$
- 目标值：$t_1, \ldots, t_N$ ，$t_n \in \{-1, 1\}$
- 新数据的分类：$\operatorname{sign}(y(\mathbf{x}))$

二维平面上线性分类问题



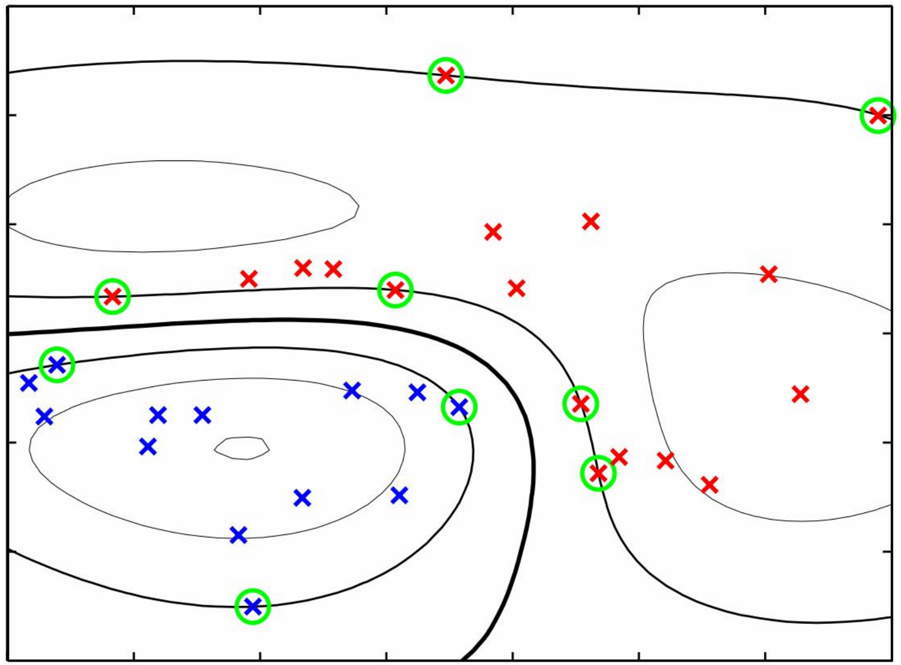
线性可分支持向量机



$\vec{w}\cdot\vec{x}+b=-1$
$\vec{w}\cdot\vec{x}+b=0$
$\vec{w}\cdot\vec{x}+b=+1$
Negative objects (y=-1)
Positive objects (y=+1)

使用(高斯)核，解决线性不可分

- 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



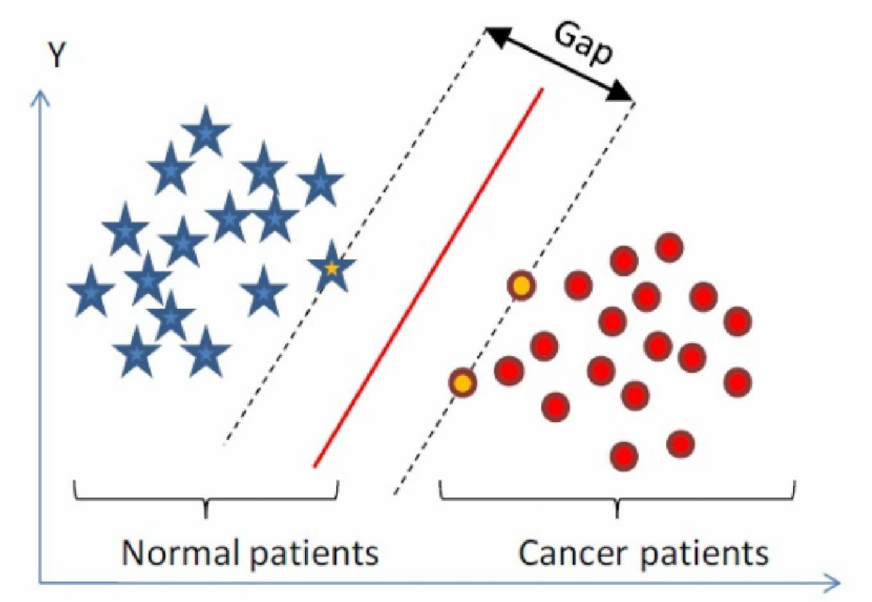
推导目标函数

- 根据题设，
$$y(\mathbf{x}_n) > 0 \Leftrightarrow t_n = +1$$
$$y(\mathbf{x}_n) < 0 \Leftrightarrow t_n = -1$$
$$\Rightarrow t_n y(\mathbf{x}_n) > 0$$

- w,b等比例缩放，则t*y的值同样缩放，从而：
$$\frac{t_n y(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{t_n(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) + b)}{\|\mathbf{w}\|}$$

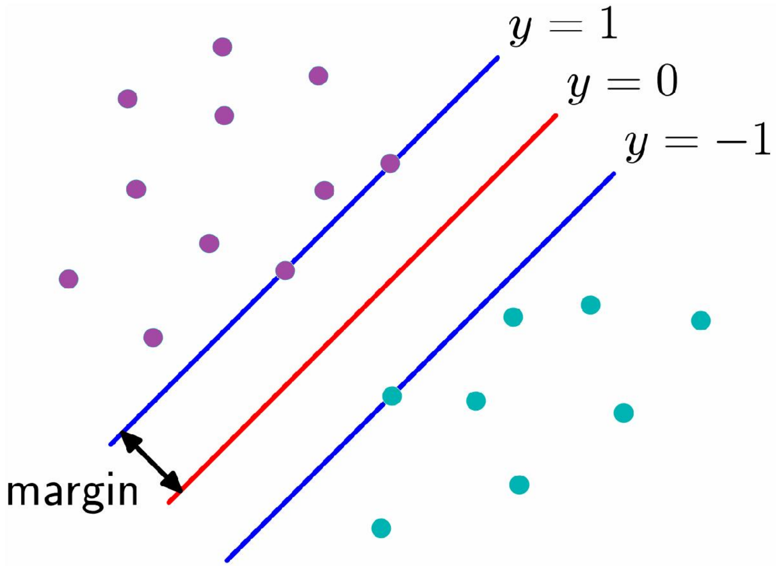
最大间隔分离超平面 $\frac{t_n y(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{t_n(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n)+b)}{\|\mathbf{w}\|}$

- 目标函数：$\arg\max_{\mathbf{w},b}\left\{\frac{1}{\|\mathbf{w}\|}\min_n\left[t_n\left(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n)+b\right)\right]\right\}$



函数间隔和几何间隔 $\frac{y(\mathbf{x}_n)}{\|\mathbf{w}\|}$

- 目标函数： $y(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) + b$
- 总可以通过等比例缩放w的方法，使得两类点的函数值都满足$|y| \geqslant 1$



建立目标函数

- 总可以通过等比例缩放w的方法，使得两类点的函数值都满足$|y| \geqslant 1$
- 约束条件：$t_n\left(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n)+b\right)=1$
- 原目标函数：

$$\arg\max_{\mathbf{w},b}\left\{\frac{1}{\|\mathbf{w}\|}\min_n\left[t_n\left(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n)+b\right)\right]\right\}$$

- 新目标函数：

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

建立目标函数：(若不考虑核函数)

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

$$s.t. \quad y_i\left(w \cdot x_i + b\right) \geq 1, \quad i = 1,2,\cdots N$$

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

$$s.t. \quad y_i\left(w \cdot x_i + b\right) \geq 1, \quad i = 1,2,\cdots N$$

拉格朗日乘子法

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N} a_n \left\{t_n(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) + b) - 1\right\}$$

- 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

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

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N} a_n \left\{t_n(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) + b) - 1\right\}$$

- 将拉格朗日函数L(w,b,**a**)分别对w，b求偏导并令其为0：

$$\frac{\partial \mathcal{L}}{\partial w} = 0 \Rightarrow \quad \mathbf{w} = \sum_{n=1}^{N} a_n t_n \phi(\mathbf{x}_n)$$

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

$$L(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{n=1}^{N} a_n \left\{ t_n(\mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_n) + b) - 1 \right\}$$

- 代入L(w,b, $\boldsymbol{\alpha}$)中，得到：

$$\begin{aligned}
\mathcal{L}(\boldsymbol{w}, b, \boldsymbol{\alpha}) &= \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n} \alpha_i \left[y_i \left(\boldsymbol{w}^{\mathrm{T}} \boldsymbol{x}_i + b \right) - 1 \right] \\
&= \frac{1}{2} \boldsymbol{w}^{\mathrm{T}} \boldsymbol{w} - \boldsymbol{w}^{\mathrm{T}} \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i - b \sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i \\
&= \frac{1}{2} \boldsymbol{w}^{\mathrm{T}} \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i - \boldsymbol{w}^{\mathrm{T}} \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i - b \cdot 0 + \sum_{i=1}^{n} \alpha_i \\
&= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \left(\sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i \right)^{\mathrm{T}} \sum_{i=1}^{n} \alpha_i y_i \boldsymbol{x}_i \\
&= \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j
\end{aligned}$$

$$\boxed{\widetilde{L}(\mathbf{a}) = \sum_{n=1}^{N} a_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a_n a_m t_n t_m k(\mathbf{x}_n, \mathbf{x}_m)}$$

继续求$\min_{w,b} L(w,b,\alpha)$对α的极大

$$\max_{\boldsymbol{\alpha}} \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^{\mathrm{T}} \boldsymbol{x}_j$$

$$s.t.\ \alpha_i \geq 0, i = 1, 2, \cdots, n$$

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

整理目标函数：添加负号

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

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

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

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

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

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

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

- 求得最优解α*

线性可分支持向量机学习算法

- 计算

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

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

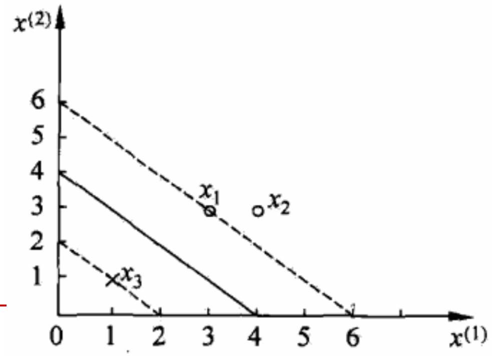
- 求得分离超平面

$$w^* x + b^* = 0$$

- 分类决策函数

$$f(x) = sign\left(w^* x + b^*\right)$$

举例



- 给定3个数据点：正例点$x_1=(3,3)^T$，$x_2=(4,3)^T$，负例点$x_3=(1,1)^T$，求线性可分支持向量机。
- 目标函数：

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

$$=\frac{1}{2}\left(18\alpha_1^2+25\alpha_2^2+2\alpha_3^2+42\alpha_1\alpha_2-12\alpha_1\alpha_3-14\alpha_2\alpha_3\right)-\alpha_1-\alpha_2-\alpha_3$$

$$s.t.\ \ \alpha_1+\alpha_2-\alpha_3=0$$

$$\alpha_i\geq 0,\quad i=1,2,3$$

将约束带入目标函数，化简计算

- 将 $\alpha_1+\alpha_2=\alpha_3$
- 带入目标函数，得到关于 α_1，α_2 的函数：

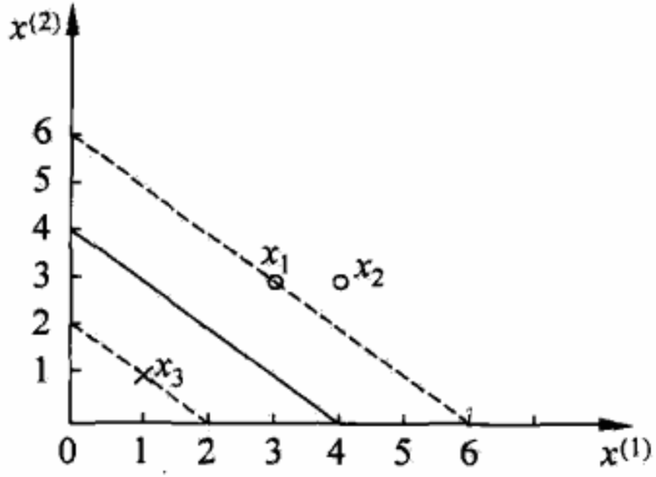
$$s(\alpha_1,\alpha_2)=4\alpha_1^2+\frac{13}{2}\alpha_2^2+10\alpha_1\alpha_2-2\alpha_1-2\alpha_2$$

- 对 α_1，α_2 求偏导并令其为0，易知 $s(\alpha_1,\alpha_2)$ 在点(1.5,-1)处取极值。而该点不满足条件 $\alpha_2\geqslant 0$，所以，最小值在边界上达到。
- 当 $\alpha_1=0$ 时，最小值 s(0,2/13)=-2/13=-0.1538
- 当 $\alpha_2=0$ 时，最小值 s(1/4,0)=-1/4=-0.25
- 于是，$s(\alpha_1,\alpha_2)$ 在 $\alpha_1=1/4$，$\alpha_2=0$ 时达到最

分离超平面

- $\alpha_1=\alpha_3=1/4$对应的点x_1,x_3是支持向量。
- 带入公式：$w^*=\sum_{i=1}^{N}\alpha_i^* y_i x_i$

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



- 得到$w_1=w_2=0.5$，$b=-2$
- 因此，分离超平面为 $\frac{1}{2}x_1+\frac{1}{2}x_2-2=0$
- 分离决策函数为

$$f(x)=sign\left(\frac{1}{2}x_1+\frac{1}{2}x_2-2\right)$$

线性支持向量机

- 若数据线性不可分，则增加松弛因子 $\xi_i \geqslant 0$，使函数间隔加上松弛变量大于等于1。这样，约束条件变成

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

- 目标函数：

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

线性SVM的目标函数

$$\min_{w,b,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i$$

$$s.t.\quad y_i\left(w\cdot x_i + b\right) \geq 1-\xi_i,\quad i=1,2,\cdots N$$

$$\xi_i \geq 0,\quad i=1,2,\cdots N$$

拉格朗日函数

- 拉格朗日函数

$$L(w,b,\xi,\alpha,\mu) \equiv \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i\left(y_i\left(w\cdot x_i + b\right) - 1 + \xi_i\right) - \sum_{i=1}^{N}\mu_i\xi_i$$

- 对w,b, ξ 求偏导

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{w}} = 0 \Rightarrow \boldsymbol{w} = \sum_{i=1}^{n}\alpha_i y_i \boldsymbol{x}_i$$

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

$$C - \alpha_i - \mu_i = 0$$

带入目标函数

- 将三式带入L中，得到

$$\min_{w,b,\xi} L(w,b,\xi,\alpha,\mu) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i \cdot x_j) + \sum_{i=1}^{N}\alpha_i$$

- 对上式求关于α的极大，得到：

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

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

$$C - \alpha_i - \mu_i = 0$$

$$\alpha_i \geq 0$$

$$\mu_i \geq 0, \quad i = 1,2,\ldots N$$

$$0 \leq \alpha_i \leq C$$

最终的目标函数

- 整理，得到对偶问题：

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

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

$$0 \le \alpha_i \le C, \qquad i = 1,2,\ldots N$$

线性支持向量机学习算法

- 构造并求解约束最优化问题

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

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

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

- 求得最优解 α *

线性支持向量机学习算法

- 计算

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

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

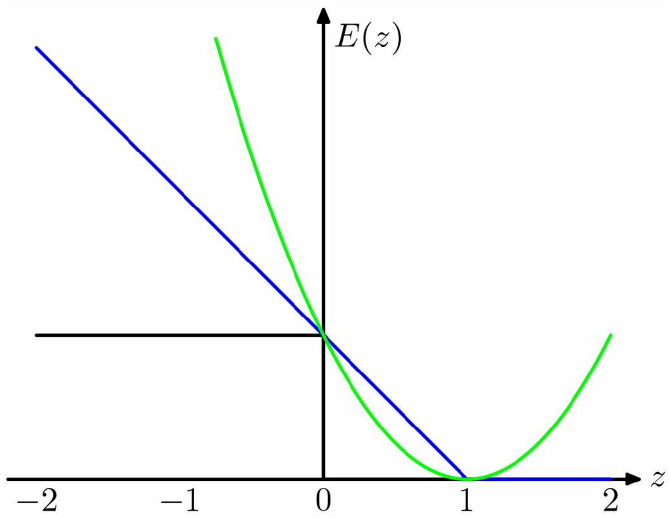
  - 注意：计算b*时，需要使用满足条件$0<\alpha_j<C$的向量
  - 实践中往往取支持向量的所有值取平均，作为b*

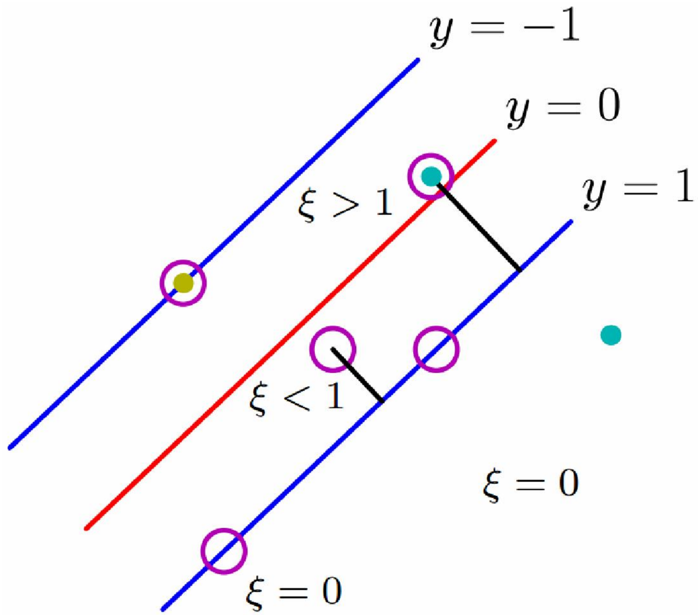
- 求得分离超平面 $w^* x + b^* = 0$
- 分类决策函数

$$f(x) = sign(w^* x + b^*)$$

损失函数分析

- 黑色：误分类率
- 蓝色：SVM合页损失
- 绿色：误差平方和





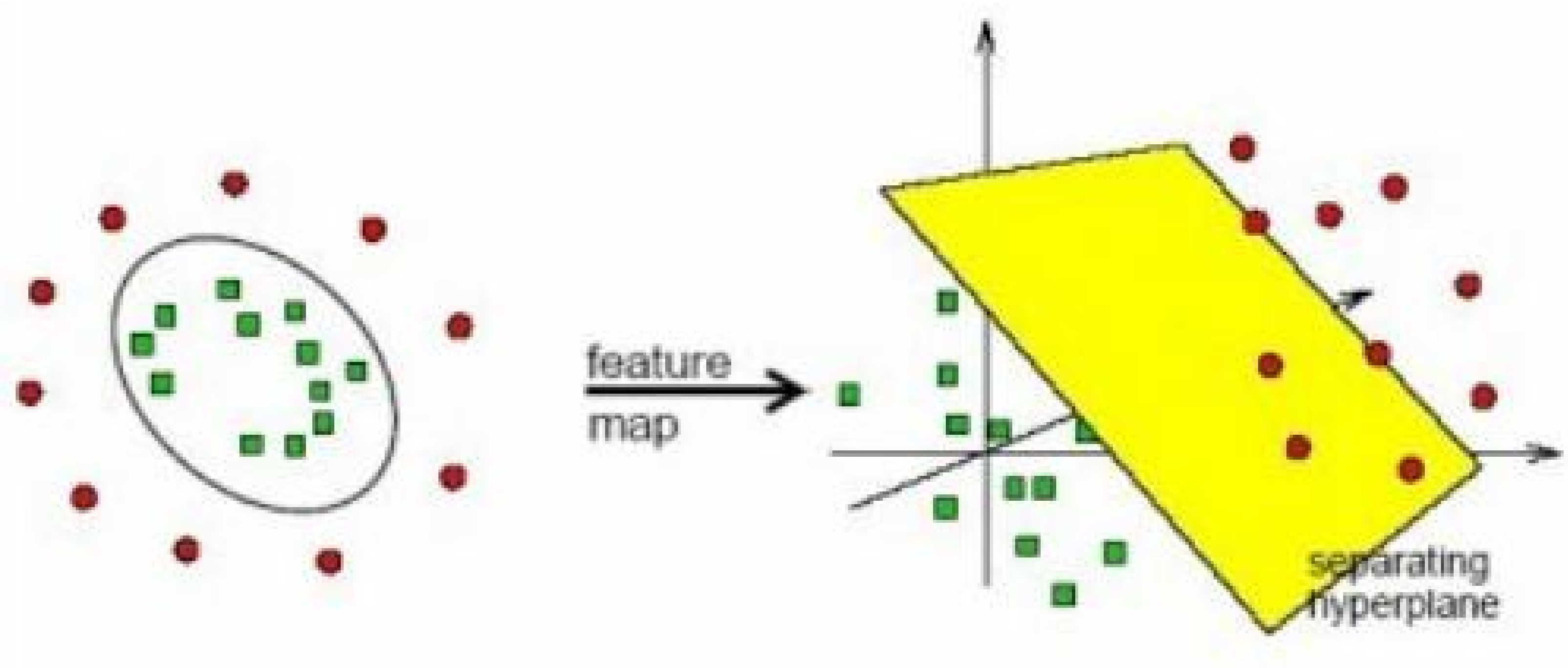
核函数

- 可以使用核函数，将输入空间映射到特征空间，从而，使得原本线性不可分的样本可以在特征空间可分。
- 在实际应用中，往往依赖先验领域知识才能选择有效的核函数
- 多项式核函数 $\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2) = (\langle \boldsymbol{x}_1, \boldsymbol{x}_2 \rangle + R)^d$
- 高斯核函数

$$\kappa(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left\{-\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\sigma^2}\right\}$$

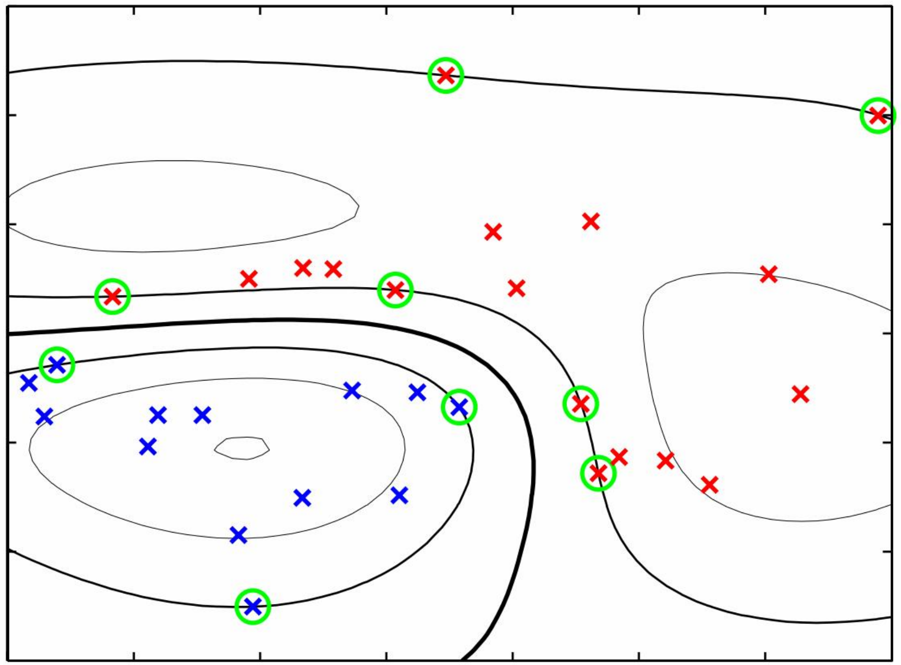
- 字符串核函数
  - 如：两个字符串的字符串编辑距离
  - 将文档使用TF-INF转换成向量，然后求向量夹角余弦

核函数映射



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



SVM中系数的求解：SMO

- 序列最小最优化
  - Sequential Minimal Optimization
- 有多个拉格朗日乘子
- 每次只选择其中两个乘子做优化，其他因子认为是常数。
  - 将N个解问题，转换成两个变量的求解问题：并且目标函数是凸的。

SMO

- 考察目标函数，假设α1和α2是变量，其他是定值：

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

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

$$0 \le \alpha_i \le C, \quad i = 1,2,\dots N$$

$$\min_{\alpha_1,\alpha_2} W(\alpha_1,\alpha_2)$$

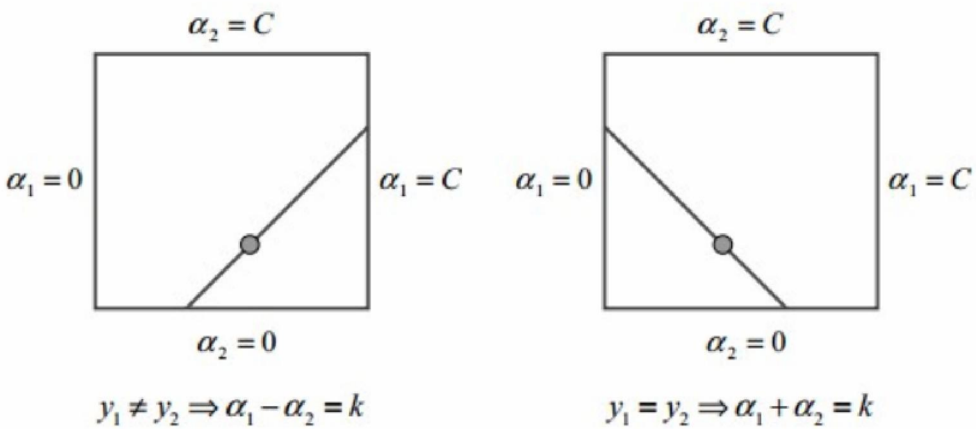
$$= \frac{1}{2}K_{11}\alpha_1^2 + \frac{1}{2}K_{22}\alpha_2^2 + y_1 y_2 \alpha_1 \alpha_2 - (\alpha_1 + \alpha_2)$$

$$+ y_1\alpha_1\sum_{i=3}^{N} y_i\alpha_i K_{i1} + y_2\alpha_2\sum_{i=3}^{N} y_i\alpha_i K_{i2}$$

$$s.t. \quad \alpha_1 y_1 + \alpha_2 y_2 = -\sum_{i=3}^{N} y_i\alpha_i = \zeta$$

$$0 \le \alpha_i \le C$$

二变量优化问题



$$\begin{cases} L = \max\{0, \alpha_j - \alpha_i\} \\ H = \max\{C, C + \alpha_j - \alpha_i\} \end{cases}, \quad y_i \neq y_j$$

$$\begin{cases} L = \max\{0, \alpha_j + \alpha_i - C\} \\ H = \max\{C, \alpha_j - \alpha_i\} \end{cases}, \quad y_i = y_j$$

SMO的迭代公式

- 迭代公式：

$$\alpha_j^{\text{new}} = \alpha_j^{\text{old}} + \frac{y_j(E_i - E_j)}{\eta}$$

$$g(x) = \sum_{i=1}^{N} y_i \alpha_i K(x_i, x) + b$$

$$E_i = g(x_i) - y_i = \left(\sum_{j=1}^{N} y_j \alpha_j K(x_j, x_i) + b \right) - y_i, \quad i = 1,2$$

SMO算法

- 1. 取初值$\alpha^{(0)}=0$，令$k=0$
- 2. 选择优化变量$\alpha_1^{(k)}$，$\alpha_2^{(k)}$，解析求解两个变量的优化问题，求得最优解$\alpha_1^{(k+1)}$，$\alpha_2^{(k+1)}$，更新α为$\alpha^{(k+1)}$
- 3. 若在精度ε范围内满足退出条件(下一页)，则转4；否则，k++，转2
- 4. 取$\alpha=\alpha^{(k+1)}$

退出条件

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

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

$$y_i \cdot g(x_i) = \begin{cases} \ge 1, & \{x_i | \alpha_i = 0\} // 落在边界外 \\ = 1, & \{x_i | 0 < \alpha_i < C\} // 落在边界上 \\ \le 1, & \{x_i | \alpha_i = C\} // 落在边界内 \end{cases}$$

$$g(x_i) = \sum_{j=1}^{N} y_j \alpha_j K(x_j, x_i) + b$$

思考

- SVM可以用来划分多类别嘛?
- SVM和Logistic回归的比较
  - 经典的SVM，直接输出类别，不给出后验概率；
  - Logistic回归，会给出属于哪个类别的后验概率。

参考文献

- Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer Press, 2006
- 统计学习方法，李航著，清华大学出版社，2012年
- Convex Optimization, Stephen Boyd,Lieven Vandenberghe, Cambridge University Press, 2004
  - 中译本：王书宁，许鋆，黄晓霖 译，凸优化，清华大学出版社，2013
- Support Vector Machines, Charlie Frogner, 2011
- Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines, John C. Platt. 1998
- Support Vector Machines, Andrew W. Moore, 2001
- http://blog.csdn.net/v_july_v/article/details/7624837

我们在这里

- 更多机器学习问题在 7 七月算法
  - http://www.julyedu.com/
    - 免费视频
    - 直播课程
    - 问答社区

- contact us：微博
  - @研究者July
  - @七月问答
  - @邹博_机器学习

感谢大家！

恳请大家批评指正！