

支持向量机

机器学习研究室

计算机科学与技术学院
吉林大学

大纲

- SVM的理论基础
- 线性判别函数和判别面
- 最优分类面
- 拉格朗日函数和对偶问题
- 软间隔最大化
- SVM分解算法
- 回归问题

SVM的理论基础

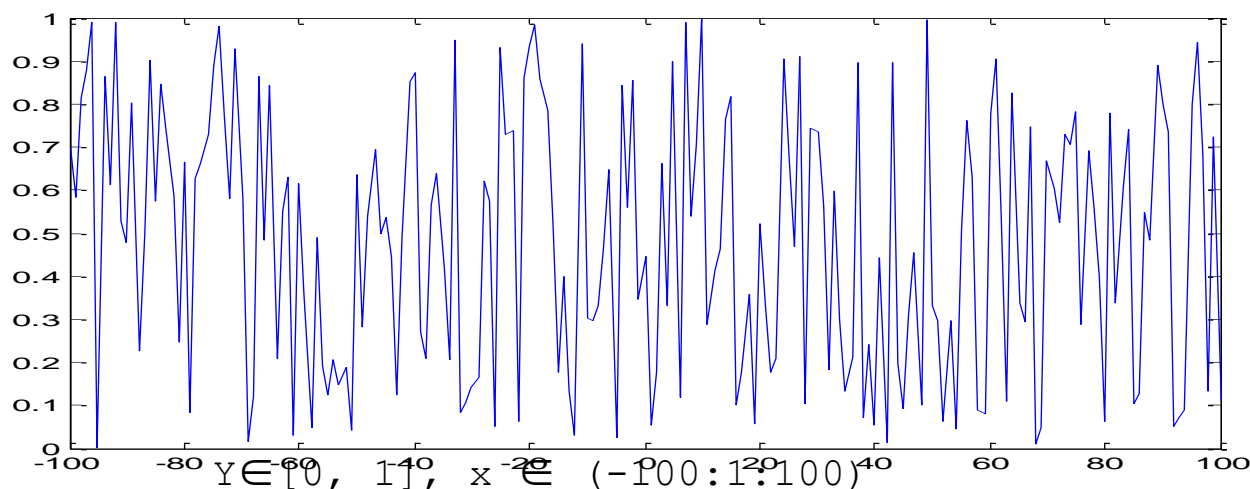
SVM的理论基础

- 传统的统计模式识别方法只有在样本趋向无穷大时，其性能才有理论上的保证。统计学习理论（SLT）研究有限样本情况下的机器学习问题。
- 统计学习理论是SVM的理论基础
- 传统的统计模式识别方法在进行机器学习时，强调经验风险最小化。而单纯的经验风险最小化会产生“过学习问题”，其推广能力较差。
- **推广能力**：学习机器（即预测函数，或称学习函数、学习模型）对未来输出进行正确预测的能力。
- **过学习问题**：学习机训练精度高，而测试精度低、推广能力差的现象。

SVM的理论基础

- 过学习问题产生的原因

例如：对一组形如 (x, y) 的训练样本, x 分布在实数范围内, y 取值在 $[0, 1]$ 之间。无论这些样本是由什么模型产生的, 我们总可以用 $y = \sin(w * x)$ 去拟合, 使得训练误差为0.



SVM的理论基础

• SVM如何克服过拟合问题

什么叫小样本？

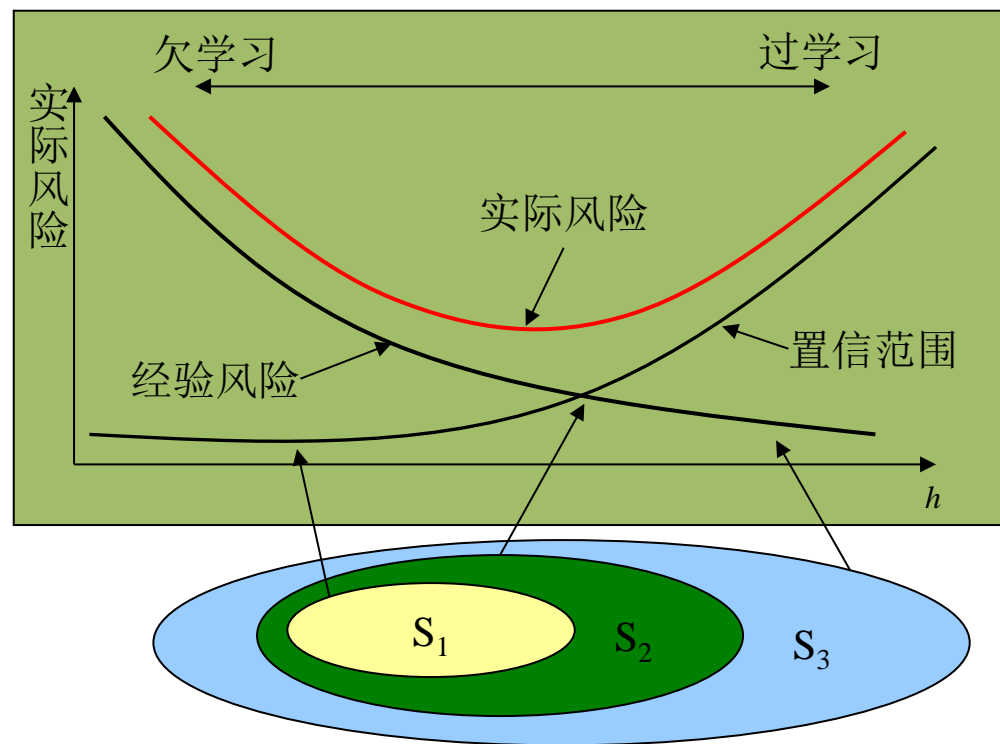


- 统计学习理论：针对小样本问题而提出的学习理论
- **VC维**：对学习函数集复杂程度的衡量，记为 h
- 经验风险：由观测样本计算的学习误差
- 期望风险：学习机在统计意义上的学习误差
- **结构风险**：在一定的VC维下，由经验风险和置信区间组成的期望风险的上届，称为结构风险
- **置信度**：与VC维和学习样本数相关的期望风险的部分上届，称为置信度

$$R(w) \leq R_{emp}(w) + \sqrt{\frac{h(\ln(2h/n) + 1) - \ln(\eta/4)}{n}} \equiv R_{emp}(w) + \Phi(h/n)$$

SVM的理论基础

- 结构风险最小化示意图



函数子集: $S_1 \subset S_2 \subset S_3$

VC维: $h_1 \leq h_2 \leq h_3$

线性判别函数和判别面

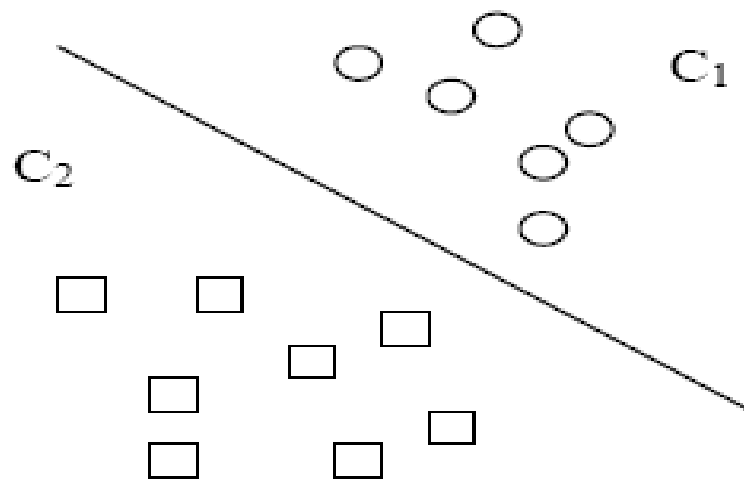
线性判别函数和判别面

- 一个线性判别函数(discriminant function)是指由 x 的各个分量的线性组合而成的函数

$$g(x) = w^T x + w_0$$

- 两类情况：对于两类问题的决策规则为

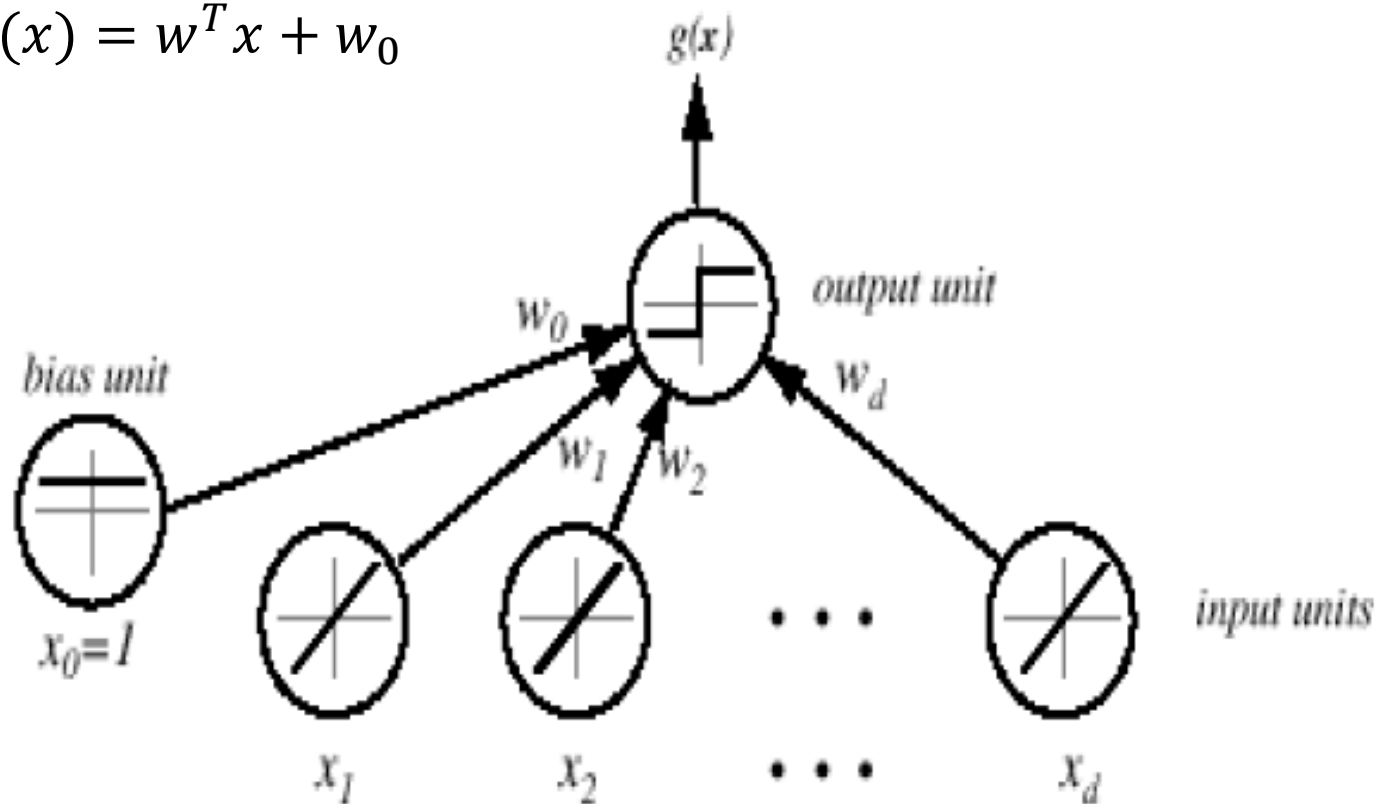
- 如果 $g(x) > 0$ ，则判定 x 属于 C_1 ，
- 如果 $g(x) < 0$ ，则判定 x 属于 C_2 ，
- 如果 $g(x) = 0$ ，则可以将 x 任意分到某一类或者拒绝判定。



线性判别函数和判别面

- 下图表示一个简单的线性分类器，具有d个输入的单元，每个对应一个输入向量在各维上的分量值。该图类似于一个神经元。

$$g(x) = w^T x + w_0$$



线性判别函数和判别面

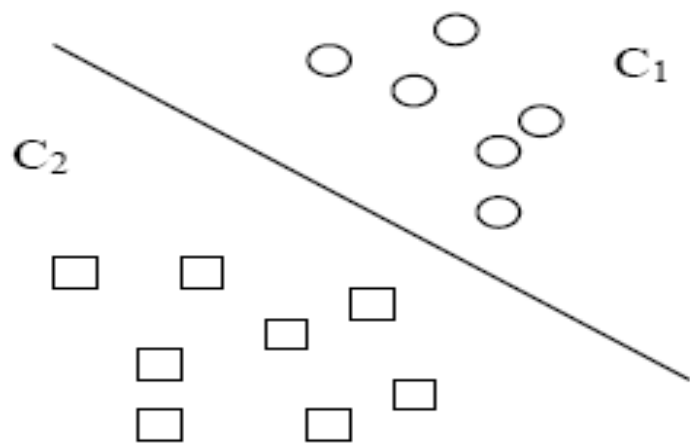
- 方程 $g(x)=0$ 定义了一个判定面，它把归类于 C_1 的点与归类于 C_2 的点分开来。
- 当 $g(x)$ 是线性函数时，这个平面被称为“超平面” (hyperplane)。
- 当 x_1 和 x_2 都在判定面上时，

$$w^T x_1 + w_0 = w^T x_2 + w_0$$

$$\text{或者 } w^T (x_1 - x_2) = 0$$

- 这表明 w 和超平面上任意向量正交，并称 w 为超平面的法向量。

注意到： $x_1 - x_2$ 表示超平面上的一个向量



线性判别函数和判别面

从下图容易看出

$$x = x_p + r \frac{w}{\|w\|}$$

将 $x = x_p + r \frac{w}{\|w\|}$ 代入 $g(x) = w^T x + w_0$ 中, 我们有

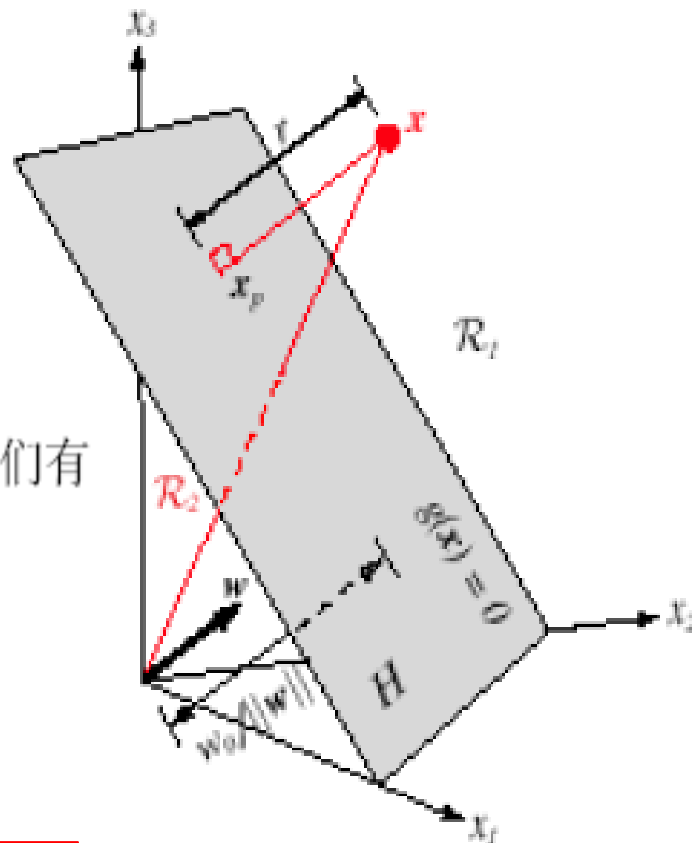
$$g(x) = w^T x + w_0$$

$$= w^T \left(x_p + r \frac{w}{\|w\|} \right) + w_0$$

$$= w^T x_p + w_0 + w^T r \frac{w}{\|w\|}$$

$$= r \|w\|$$

判别函数 $g(x)$ 是特征空间中某点 x 到超平面的距离的一种代数度量

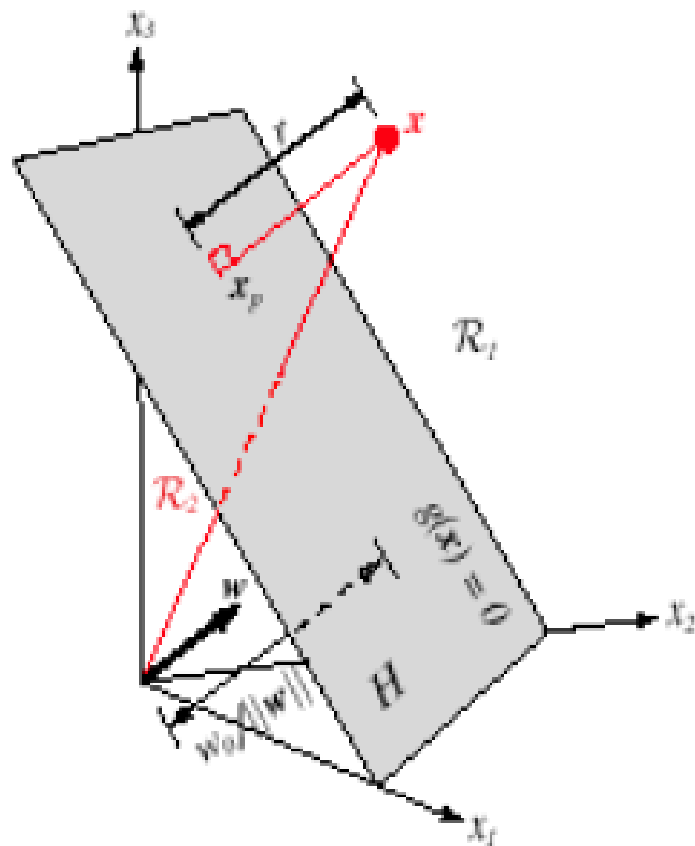


线性判别函数和判别面

- 上式也可以表示为： $r = g(x) / \|w\|$ 。当 $x=0$ 时，表示原点到超平面的距离， $r_0 = g(0) / \|w\| = w_0 / \|w\|$ ，标示在上图中。

总之：

- 线性判别函数利用一个超平面把特征空间分隔成两个区域。
- 超平面的方向由法向量 w 确定，它的位置由阈值 w_0 确定。
- 判别函数 $g(x)$ 正比于 x 点到超平面的代数距离（带正负号）。当 x 点在超平面的正侧时， $g(x) > 0$ ；当 x 点在超平面的负侧时， $g(x) < 0$



线性判别函数和判别面

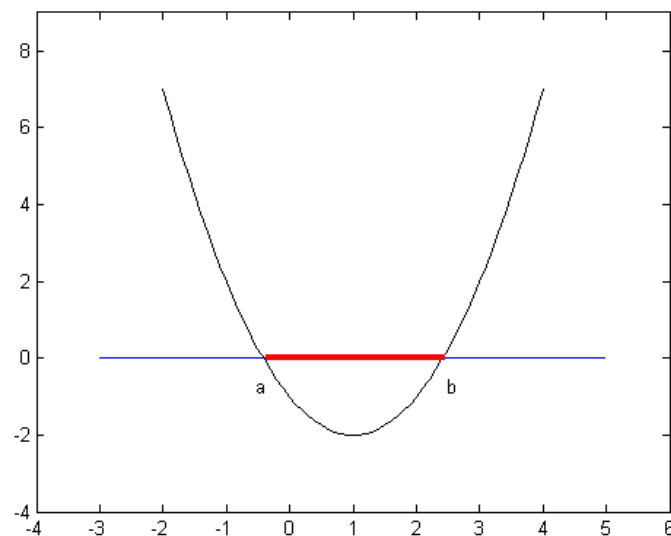
广义线性判别函数

在一维空间中，没有任何一个线性函数能解决下述划分问题（黑红各代表一类数据），可见线性判别函数有一定的局限性。



线性判别函数和判别面

- 如果建立一个二次判别函数 $g(x)=(x-a)(x-b)$ ，则可以很好的解决上述分类问题。
- 决策规则仍是：如果 $g(x)>0$ ，则判定 x 属于 C_1 ，如果 $g(x)<0$ ，则判定 x 属于 C_2 ，如果 $g(x)=0$ ，则可以将 x 任意分到某一类或者拒绝判定。



线性判别函数和判别面

二次判别函数 $g(x)=(x-a)(x-b)$ 可写成如下的一般形式:

$$g(x) = c_0 + c_1x + c_2x^2$$

如果选择 $x \rightarrow y$ 的映射, 则可以把二次判别函数化为关于 y 的线性函数

$$g(x) = a^T y = \sum_{i=1}^3 a_i y_i$$

其中,

$$y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 \\ x \\ x^2 \end{bmatrix}, \quad a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix}$$

称 $g(x) = a^T y$ 为广义线性判别函数, a 叫做广义权向量

线性判别函数和判别面

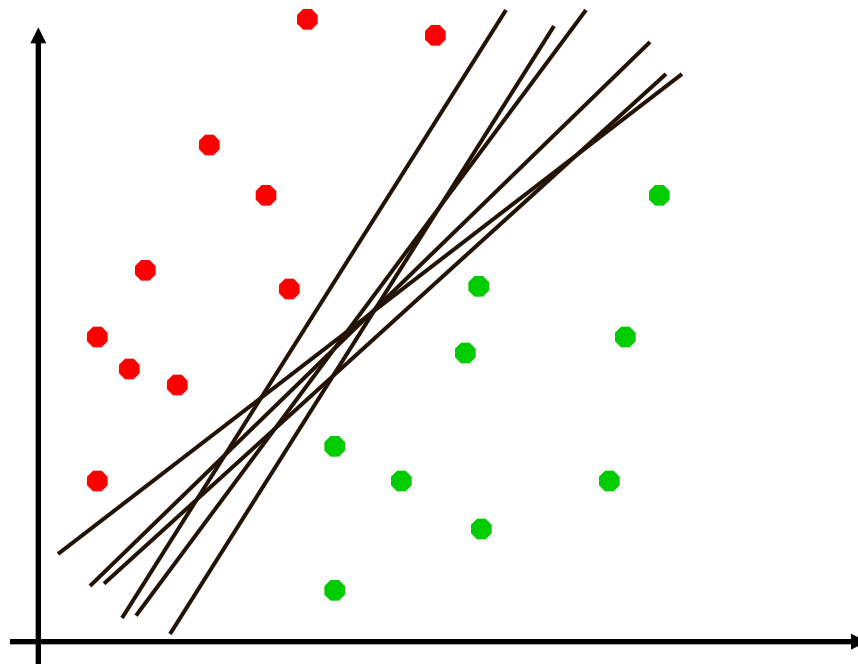
一般地，对于任意高次判别函数 $g(x)$ ，都可以通过适当的变换，化为广义线性判别函数来处理， $a^T y$ 不是 x 的线性函数，但却是 y 的线性函数。 $a^T y = 0$ 在 Y 空间确定了一个通过原点的超平面。此处的 $g(x)$ 也可以看作任意判别函数作级数展开，然后取其截尾部分的逼近。

这样我们就可以利用线性判别函数的简单性来解决复杂问题。同时带来的问题是，维数大大增加了，这将使问题很快陷入所谓的“维数灾难”。

最优分类面

最优分类面

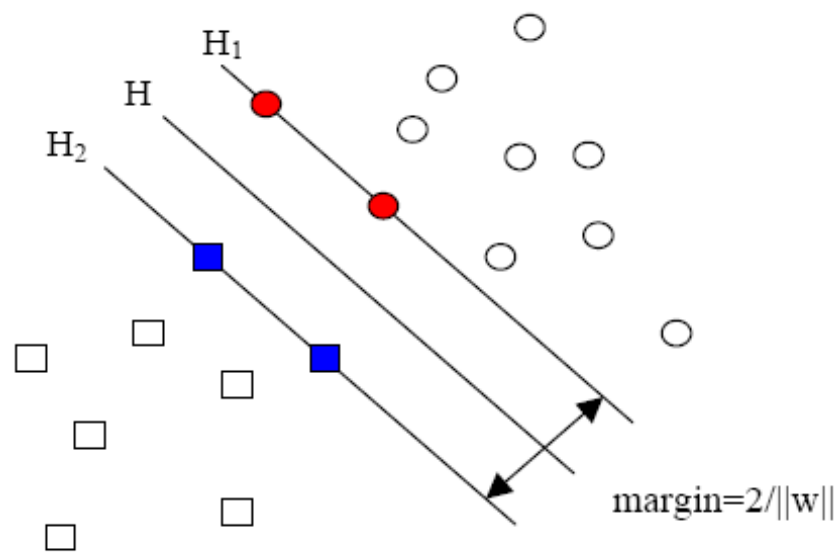
- Which of the linear separators is optimal?



最优分类面

- SVM 是从线性可分情况下的最优分类面发展而来的，基本思想可用图2的两维情况说明。

图中，方形点和圆形点代表两类样本， H 为分类线， H_1 ， H_2 分别为过各类中离分类线最近的样本且平行于分类线的直线，它们之间的距离叫做**分类间隔** (margin)。



所谓最优分类线就是要求分类线不但能将两类正确分开 (训练错误率为0)，而且使分类间隔最大。

推广到高维空间，最优分类线就变为**最优分类面**。

最优分类面

设线性可分的样本集 (x_i, y_i) , $i=1, \dots, n$, $x \in \mathbb{R}^d$, $y \in \{+1, -1\}$ 。d 维空间中的线性判别函数: $g(x) = wx + b$, 分类面方程为 $wx + b = 0$

我们可以对它进行归一化, 使得所有样本都满足 $|g(x)| \geq 1$, 即离分类面最近的样本满足 $|g(x)| = 1$, 这样分类间隔就等于 $2/||w||$ 。因此要求分类间隔最大, 就是要求 $||w||$ (或 $||w||^2$) 最小。而要求分类面对所有样本正确分类, 就是要求满足

$$y_i[wx_i + b] - 1 \geq 0, \quad i=1, \dots, n, \quad (1)$$

因此, 满足上面公式且使 $||w||^2$ 最小的分类面就是最优分类面。过两类样本中离分类面最近的点且平行于最优分类面的超平面 H_1, H_2 上的训练样本, 就是使上式等号成立的样本称作支持向量。

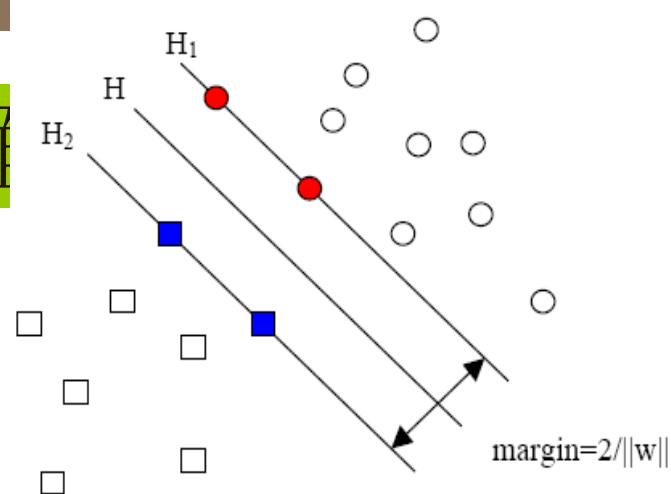


图2 线性可分情况下的最优分类线

如何求最优分类面

求最优分类面问题可以转化为如下的约束优化问题：

$$\begin{aligned} \min \quad & \frac{1}{2} w^T w \\ \text{s.t.} \quad & y_i (w^T x_i + b) - 1 \geq 0 \quad i = 1, 2, \dots, l \end{aligned}$$

这是一个二次凸优化问题，由于目标函数和约束条件都是凸的，根据最优化理论，这一问题存在唯一全局最小解。

拉格朗日函数和对偶问题

拉格朗日因子法

求 $f(\mathbf{x}) = x_1^2 + x_2^2$ 的最小值, 约束条件: $g(\mathbf{x}) = x_1^2 x_2 - 3 = 0$
 $\min f(\mathbf{x}) \quad s.t. \quad g(\mathbf{x}) = 0$

- 约束曲线与极值曲线**相切**的点为极值点 \mathbf{x}^* 。
 - 对于约束曲面上的任意点 \mathbf{x} , 该点的梯度 $\nabla g(\mathbf{x})$ 正交于约束曲面。
 - 在最优点 \mathbf{x}^* , **目标函数**在该点的梯度 $\nabla f(\mathbf{x}^*)$ 正交于约束曲面。

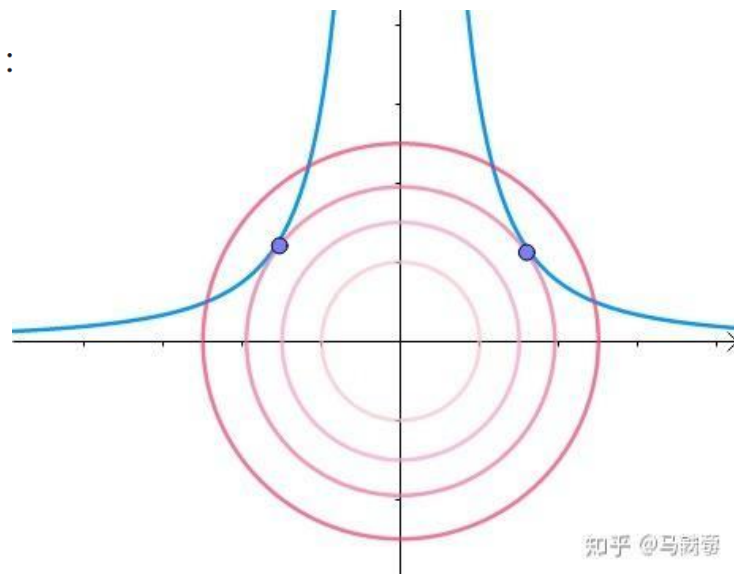
由此可知, 在最优点 \mathbf{x}^* , 梯度 $\nabla g(\mathbf{x})$ 和 $\nabla f(\mathbf{x})$ 的方向必相同或相反, 即存在 $\lambda \neq 0$, 使得: $\nabla f(\mathbf{x}^*) + \lambda \nabla g(\mathbf{x}^*) = 0$, λ 称之为**拉格朗日乘子**。

所以在求解 $f(x)$ 极值的问题上, 我们相当于有两个条件了:

$$\begin{cases} \nabla f(\mathbf{x}) + \lambda \nabla g(\mathbf{x}) = 0 \\ g(\mathbf{x}) = x_1^2 x_2 - 3 = 0 \end{cases}$$

$$\begin{cases} \nabla f(\mathbf{x}) + \lambda \nabla g(\mathbf{x}) = 0 \\ g(\mathbf{x}) = 0 \end{cases}$$

$$L(\mathbf{x}, \lambda) = f(x) + \lambda g(x)$$



拉格朗日因子法

求 $f(\mathbf{x}) = x_1^2 + x_2^2$ 的最小值, 约束条件: $g(\mathbf{x}) = x_1^2 x_2 - 3 = 0$
 $\min f(\mathbf{x}) \quad s.t. \quad g(\mathbf{x}) = 0$

$$\begin{cases} \nabla f(\mathbf{x}) + \lambda \nabla g(\mathbf{x}) = 0 \\ g(\mathbf{x}) = 0 \end{cases}$$

$$L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x})$$

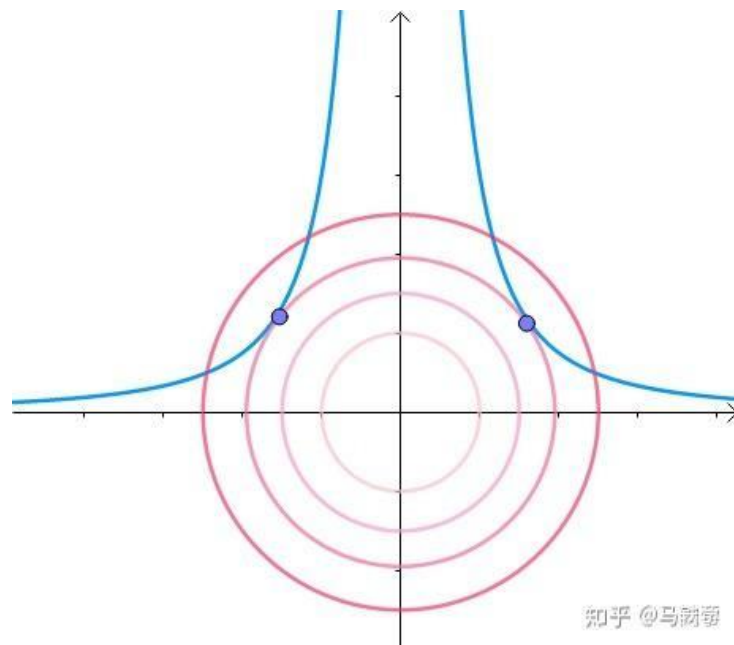
将其对 \mathbf{x} 的偏导数 $\nabla_{\mathbf{x}} L(\mathbf{x}, \lambda)$ 置零,

$$\nabla f(\mathbf{x}) + \lambda \nabla g(\mathbf{x}) = 0$$

将其对 λ 的偏导数 $\nabla_{\lambda} L(\mathbf{x}, \lambda)$ 置零

$$g(\mathbf{x}) = 0$$

这就是在求 $L(\mathbf{x}, \lambda)$ 的极值点



拉格朗日因子法

不等式约束:

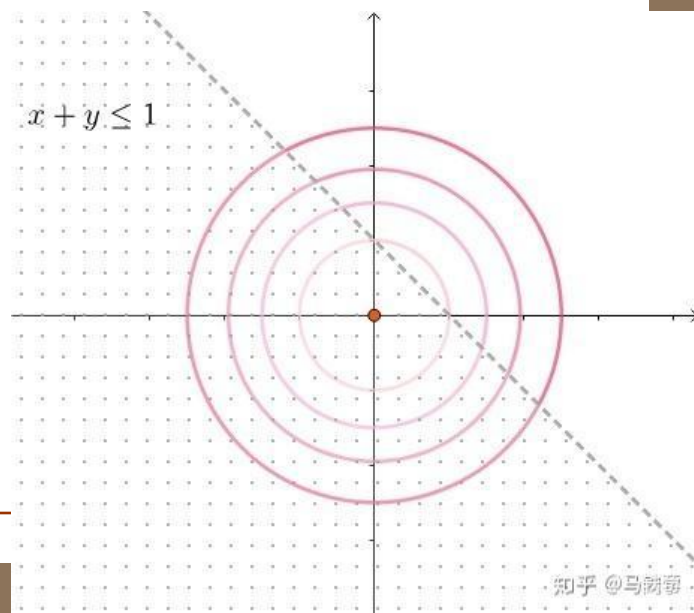
最优解的位置只有两种情况，一种是最优解在不等式约束的**边界上**，另一种就是**不等式约束的区域内**，需要分为两种情况讨论。例子：

情况一：最优点在不等式约束的区域内

- 求 $f(\mathbf{x}) = x_1^2 + x_2^2$ 的最小值，约束条件： $g(\mathbf{x}) = x_1 + x_2 - 1 \leq 0$
- $\min f(\mathbf{x}) \quad s.t. \quad g(\mathbf{x}) \leq 0$

可以看到，这个不等式约束实际上包含了原点，所以这个约束等于没有。按照正常求极值办法，直接对 $f(\mathbf{x})$ 求梯度，令其等于0，即可得到极值点。

$$\nabla f(\mathbf{x}) = 0 \Rightarrow (x_1, x_2) = (0, 0)$$



拉格朗日因子法

情况二：最优点在不等式的边界上

- 求 $f(\mathbf{x}) = x_1^2 + x_2^2$ 的最小值, 约束条件: $g(\mathbf{x}) = x_1 + x_2 + 2 \leq 0$
- $\min f(\mathbf{x}) \quad s.t. g(\mathbf{x}) \leq 0$

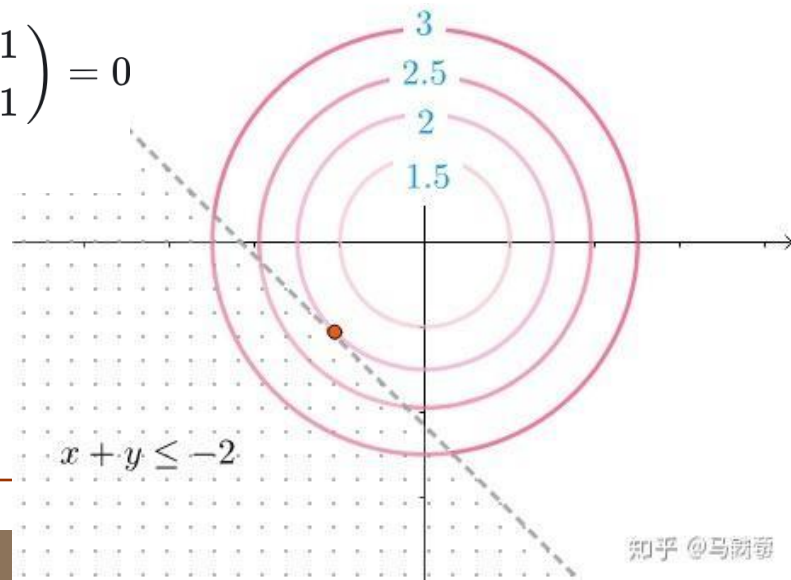
可以看到, 在不等式约束下, 最优解是在边缘相切的地方取得, 换句话说, 最优解也在约束的边界上。此时等价于等式约束。

$$\min f(\mathbf{x}) \quad s.t. g(\mathbf{x}) = 0$$

写出拉格朗日函数: $L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x}) = x_1^2 + x_2^2 + \lambda(x_1 + x_2 + 2)$, $\lambda \neq 0$, 现在求约束条件下的极值问题就转换成求 $L(\mathbf{x}, \lambda)$ 的极值问题啦, 所以令 $\nabla_x L(\mathbf{x}, \lambda) = 0$ 和 $\nabla_\lambda L(\mathbf{x}, \lambda) = 0$ 可求得最优解。

$$\begin{cases} \nabla_x L(\mathbf{x}, \lambda) = \nabla f(\mathbf{x}) + \lambda \nabla g(\mathbf{x}) = \begin{pmatrix} 2x_1 \\ 2x_2 \end{pmatrix} + \lambda \begin{pmatrix} 1 \\ 1 \end{pmatrix} = 0 \\ \nabla_\lambda L(\mathbf{x}, \lambda) = g(\mathbf{x}) = x_1 + x_2 + 2 = 0 \end{cases}$$

$$\Rightarrow \begin{cases} x_1 = -1 \\ x_2 = -1 \\ \lambda = 0.5 \end{cases}$$



拉格朗日因子法

两种情况合并：

- **情况一：**最优解在 $g(\mathbf{x}) < 0$ 区域内，条件无作用，直接令 $\nabla f(\mathbf{x}) = 0$ 求解。
 - 这等价于将拉格朗日函数的 λ 置零，即 $L(\mathbf{x}, \lambda) = f(\mathbf{x}) + \lambda g(\mathbf{x}) = f(\mathbf{x})$ ，再对 $L(\mathbf{x}, \lambda)$ 求极值。
- **情况二：**最优解在 $g(\mathbf{x}) = 0$ 上，相当于等式约束，曲线相切处为最优解。
 - 即满足 存在一个 λ 使得 $\nabla f(x^*) + \lambda \nabla g(x^*) = 0$ ，但是这里的 λ 的取值范围就不是不等于0了，这个时候最优解处 $\nabla f(x^*)$ 的方向必须与 $\nabla g(x^*)$ 的**相反**，即存在常数 $\lambda > 0$ ，使得 $\nabla f(x^*) + \lambda \nabla g(x^*) = 0$ 。

因此，不论是内部解或边界解， $\lambda g(\mathbf{x}) = 0$ 恒成立，称为**互补松弛性(complementary slackness)**。整合上述两种情况，最佳解的必要条件包括Lagrangian函数 $L(\mathbf{x}, \lambda)$ 的**定常方程式、原始可行性、对偶可行性，以及互补松弛性**：

$$\left\{ \begin{array}{l} \nabla_{\mathbf{x}} L = \nabla f + \lambda \nabla g = \mathbf{0} \\ g(\mathbf{x}) \leq 0 \\ \lambda \geq 0 \\ \lambda g(\mathbf{x}) = 0. \end{array} \right.$$

这些条件合称为Karush-Kuhn-Tucker (KKT)条件。如果我们要最大化 $f(\mathbf{x})$ 且受限于 $g(\mathbf{x}) \leq 0$ ，那么对偶可行性要改成 $\lambda \leq 0$ 。

拉格朗日因子法

$$\begin{aligned} \min & \frac{1}{2} w^T w \\ \text{s.t. } & y_i (w^T x_i + b) - 1 \geq 0 \quad i = 1, 2, \dots, l \end{aligned}$$

- 线性支持向量机的拉格朗日函数为

$$L(w, b, \alpha) = \frac{1}{2} w^T w + \sum_{i=1}^l \alpha_i [1 - y_i (w^T x_i + b)]$$

KKT条件

$$\frac{\partial L(w, b, \alpha)}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^l \alpha_i y_i x_i$$

$$\frac{\partial L(w, b, \alpha)}{\partial b} = 0 \Rightarrow \sum_{i=1}^l \alpha_i y_i = 0$$

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

$$\alpha_i [y_i (w^T x_i + b) - 1] = 0 \quad i = 1, 2, \dots, l$$

前两式代入拉格朗日函数

$$L(w, b, \alpha) = -\frac{1}{2} w^T w + \sum_{i=1}^l \alpha_i = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{p=1}^l \sum_{i=1}^l \alpha_p \alpha_i y_p y_i (x_p)^T x_i \equiv Q(\alpha)$$

对偶问题

定义 3 (拉格朗日函数). 对于优化问题

$$\begin{aligned} \min_{\mathbf{u}} \quad & f(\mathbf{u}) \\ \text{s. t.} \quad & g_i(\mathbf{u}) \leq 0, \quad i = 1, 2, \dots, m, \\ & h_j(\mathbf{u}) = 0, \quad j = 1, 2, \dots, n, \end{aligned}$$

定义其拉格朗日函数为

$$\mathcal{L}(\mathbf{u}, \boldsymbol{\alpha}, \boldsymbol{\beta}) := f(\mathbf{u}) + \sum_{i=1}^m \alpha_i g_i(\mathbf{u}) + \sum_{j=1}^n \beta_j h_j(\mathbf{u}), \quad (20)$$

其中 $\alpha_i \geq 0$.

引理 7. 公式 19 描述的优化问题等价于

$$\begin{aligned} \min_{\mathbf{u}} \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \quad & \mathcal{L}(\mathbf{u}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \\ \text{s. t.} \quad & \alpha_i \geq 0, \quad i = 1, 2, \dots, m. \end{aligned} \quad (21)$$

$$\begin{aligned} & \min_{\mathbf{u}} \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \mathcal{L}(\mathbf{u}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \\ (19) \quad & = \min_{\mathbf{u}} \left(f(\mathbf{u}) + \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \left(\sum_{i=1}^m \alpha_i g_i(\mathbf{u}) + \sum_{j=1}^n \beta_j h_j(\mathbf{u}) \right) \right) \\ & = \min_{\mathbf{u}} \left(f(\mathbf{u}) + \begin{cases} 0 & \text{若 } \mathbf{u} \text{ 满足约束;} \\ \infty & \text{否则} \end{cases} \right) \\ & = \min_{\mathbf{u}} f(\mathbf{u}), \text{ 且 } \mathbf{u} \text{ 满足约束,} \end{aligned} \quad (22)$$

其中, 当 g_i 不满足约束时, 即 $g_i(\mathbf{u}) > 0$, 我们可以取 $\alpha_i = \infty$, 使得 $\alpha_i g_i(\mathbf{u}) = \infty$; 当 h_j 不满足约束时, 即 $h_j(\mathbf{u}) \neq 0$, 我们可以取 $\beta_j = \text{sign}(h_j(\mathbf{u}))\infty$, 使得 $\beta_j h_j(\mathbf{u}) = \infty$. 当 \mathbf{u} 满足约束时, 由于 $\alpha_i \geq 0$, $g_i(\mathbf{u}) \leq 0$, 则 $\alpha_i g_i(\mathbf{u}) \leq 0$. 因此 $\alpha_i g_i(\mathbf{u})$ 最大值为 0.

对偶问题

- 对偶问题

$$\max_{\alpha, \beta} \min_{\mathbf{u}} \mathcal{L}(\mathbf{u}, \alpha, \beta) \quad (23)$$

$$\text{s. t.} \quad \alpha_i \geq 0, \quad i = 1, 2, \dots, m.$$

引理 9. 对偶问题是主 (primal) 问题的下界, 即

$$\max_{\alpha, \beta} \min_{\mathbf{u}} \mathcal{L}(\mathbf{u}, \alpha, \beta) \leq \min_{\mathbf{u}} \max_{\alpha, \beta} \mathcal{L}(\mathbf{u}, \alpha, \beta). \quad (24)$$

证明. 对任意 (α', β') , $\min_{\mathbf{u}} \mathcal{L}(\mathbf{u}, \alpha', \beta') \leq \min_{\mathbf{u}} \max_{\alpha, \beta} \mathcal{L}(\mathbf{u}, \alpha, \beta)$. 当 $(\alpha', \beta') = \max_{\alpha', \beta'} \min_{\mathbf{u}} \mathcal{L}(\mathbf{u}, \alpha', \beta')$ 时, 该式仍然成立, 即 $\max_{\alpha', \beta'} \min_{\mathbf{u}} \mathcal{L}(\mathbf{u}, \alpha', \beta') \leq \min_{\mathbf{u}} \max_{\alpha, \beta} \mathcal{L}(\mathbf{u}, \alpha, \beta)$.

引理 (Slater 条件). 当主问题为凸优化问题, 即 f 和 g_i 为凸函数, h_j 为仿射函数, 且可行域中至少有一点使不等式约束严格成立时, 对偶问题等价于原问题。

$$\mathcal{L}(\mathbf{w}, b, \alpha) := \frac{1}{2} \mathbf{w}^\top \mathbf{w} + \sum_{i=1}^m \alpha_i (1 - y_i (\mathbf{w}^\top \mathbf{x}_i + b)). \quad (25)$$

其对偶问题为

$$\max_{\alpha} \min_{\mathbf{w}, b} \frac{1}{2} \mathbf{w}^\top \mathbf{w} + \sum_{i=1}^m \alpha_i (1 - y_i (\mathbf{w}^\top \mathbf{x}_i + b)) \quad (26)$$

$$\text{s. t.} \quad \alpha_i \geq 0, \quad i = 1, 2, \dots, m.$$

对偶问题

原始优化问题

$$\begin{aligned} & \min \frac{1}{2} w^T w \\ & \text{s.t. } y_i (w^T x_i + b) - 1 \geq 0 \quad i = 1, 2, \dots, l \end{aligned}$$

拉格朗日函数

$$L(w, b, a) = \frac{1}{2} w^T w + \sum_{i=1}^l a_i [1 - y_i (w^T x_i + b)]$$

原问题等价于

$$\min_{w, b} \max_a L(w, b, a)$$

由于原问题为凸优化问题，故其等价于其对偶问题

$$\max_a \min_{w, b} L(w, b, a)$$

对偶问题

$$\max_a \min_{w,b} L(w, b, a)$$

因为公式内层对 (w, b) 的优化属于无约束优化问题，我们可以通过令偏导等于零的方法得到 (w, b) 的最优值。

$$\frac{\partial L(w, b, \alpha)}{\partial w} = 0 \Rightarrow w = \sum_{i=1}^l \alpha_i y_i x_i$$

$$\frac{\partial L(w, b, \alpha)}{\partial b} = 0 \Rightarrow \sum_{i=1}^l \alpha_i y_i = 0$$

代入拉格朗日函数

$$\begin{aligned} L(w, b, \alpha) &= -\frac{1}{2} w^T w + \sum_{i=1}^l \alpha_i \\ &= \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{p=1}^l \sum_i \alpha_p \alpha_i y_p y_i (x_p)^T x_i \equiv Q(\alpha) \end{aligned}$$

原问题等价于

$$\max_a Q(\alpha)$$

对偶问题

$$\begin{aligned} L(w, b, \alpha) &= -\frac{1}{2} w^T w + \sum_{i=1}^l \alpha_i \\ &= \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{p=1}^l \sum_i^l \alpha_p \alpha_i y_p y_i (x_p)^T x_i \equiv Q(\alpha) \end{aligned}$$

则问题转化为极大化函数 $Q(a)$ 的“对偶”问题。即对给定训练样本，求使 $Q(a)$ 最大的Lagrange系数并满足约束和KKT条件

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

$$\alpha_i [y_i (w^T x_i + b) - 1] = 0 \quad i = 1, 2, \dots, l$$

这是一个二次规划问题，可使用通用的二次规划算法求解；然而，该问题的规模正比于训练样本数，这会在实际任务中造成很大的开销。为了避开这个障碍，人们通过利用问题本身的特性，提出了很多高效算法，SMO(Sequential Minimal Optimization)是其中一个著名的代表。

最优分类面

- 设解为 $\{\alpha_{01}, \alpha_{02}, \dots, \alpha_{0l}\}$, 则

$$w^* = \sum_{i=1}^l \alpha_{0i} y_i x_i = \sum_{i \in SV} \alpha_{0i} y_i x_i$$

其中 SV 代表所有支持向量的集合, b 可以由互补松弛算出。对于某一支持向量 x_s 及其标记 y_s , 由于

$$y_s(w^T x_s + b) = 1, \text{ 则}$$

$$b = y_s - w^T x_s = y_s - \sum_{i \in SV} \alpha_i y_i x_i^T x_s.$$

实践中, 为了得到对 b 更稳健的估计, 通常使用对所有支持向量求解得到 b 的平均值。

最后可得到解上述问题的最优分类函数为:

$$f(x) = \text{sgn}\{w^* \cdot x + b^*\} = \text{sgn}\left\{\sum_{i=1}^k \alpha_i^* y_i (x_i \cdot x) + b^*\right\}$$

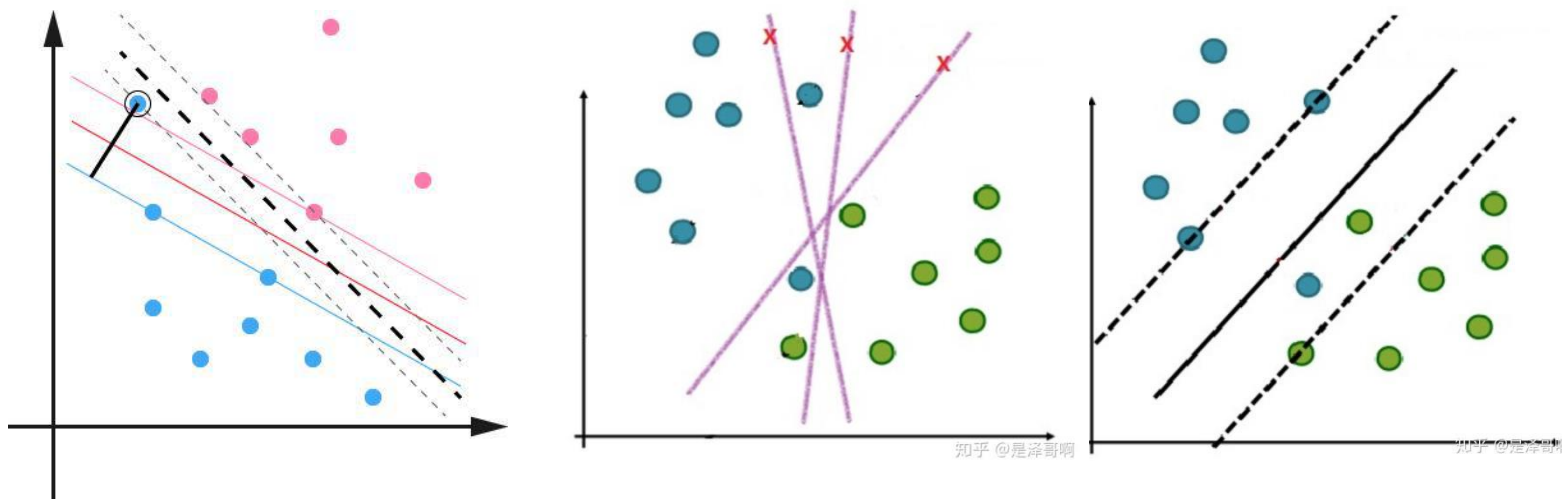
其中 α^* , b^* 为确定最优划分超平面的参数, $(x_i \cdot x)$ 为两个向量的点积。

由式(5) 知: 非支持向量对应的 α_i 都为零, 求和只对少数支持向量进行。



软间隔最大化

软间隔最大化



如果数据集中存在噪点的话，那么在求超平的时候就会出现很大问题。从下图中可以看出其中一个蓝点偏差太大，如果把它作为支持向量的话所求出来的margin就会比不算入它时要小得多。更糟糕的情况是如果这个蓝点落在了红点之间那么就找不出超平面了。因此引入一个松弛变量 ξ 来允许一些数据可以处于分隔面错误的一侧。这时新的约束条件变为：

最常用的是hinge损失：

$$l_{\text{hinge}}(z) = \max(0, 1 - z)$$

$$\min_w \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \xi_i$$

$$\min_{W, b} \frac{1}{2} \|W\|^2 + C \sum_{i=1}^n \max(0, 1 - y_i(X_i^T W + b)) \quad s.t. \quad g_i(w, b) = 1 - y_i(w^T x_i + b) - \xi_i \leq 0, \quad \xi_i \geq 0, \quad i = 1, 2, \dots, n$$

软间隔最大化

$$\min_w \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \xi_i$$
$$s. t. \quad g_i(w, b) = 1 - y_i(w^T x_i + b) - \xi_i \leq 0, \quad \xi_i \geq 0, \quad i = 1, 2, \dots, n$$

构造拉格朗日函数：

$$\min_{w, b, \xi} \max_{\lambda, \mu} L(w, b, \xi, \lambda, \mu) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \xi_i + \sum_{i=1}^n \lambda_i [1 - \xi_i - y_i(w^T x_i + b)] -$$
$$\sum_{i=1}^n \mu_i \xi_i$$
$$s. t. \quad \lambda_i \geq 0 \quad \mu_i \geq 0$$

其中 λ_i 和 μ_i 是拉格朗日乘子⁺， w 、 b 和 ξ_i 是主问题参数。

根据强对偶性，将对偶问题转换为：

$$\max_{\lambda, \mu} \min_{w, b, \xi} L(w, b, \xi, \lambda, \mu)$$

分别对主问题参数 w 、 b 和 ξ_i 求偏导数，并令偏导数为 0，得出如下关系：

$$w = \sum_{i=1}^m \lambda_i y_i x_i$$

$$0 = \sum_{i=1}^m \lambda_i y_i$$

$$C = \lambda_i + \mu_i$$

将这些关系带入拉格朗日函数中，得到：

$$\min_{w, b, \xi} L(w, b, \xi, \lambda, \mu) = \sum_{j=1}^n \lambda_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j y_i y_j (x_i \cdot x_j)$$

最小化结果只有 λ 而没有 μ ，所以现在只需要最大化 λ 就好：

$$\begin{aligned} \max_{\lambda} & \left[\sum_{j=1}^n \lambda_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j y_i y_j (x_i \cdot x_j) \right] \\ \text{s.t.} & \sum_{i=1}^n \lambda_i y_i = 0, \quad \lambda_i \geq 0, \quad C - \lambda_i - \mu_i = 0 \end{aligned}$$

我们可以看到这个和硬间隔的一样，只是多了个约束条件。

然后我们利用 SMO 算法求解得到拉格朗日乘子 λ^* 。

软间隔最大化

- Lagrange函数

$$L(w, b, \xi, \lambda, \mu) = \frac{1}{2} w^T w + C \sum_{i=0}^m \xi_i + \sum_{i=0}^n \lambda_i [1 - \xi_i - y_i(w^T x_i + b)] - \sum_{i=0}^m \mu_i \xi_i$$

则其KKT条件为

$$\begin{cases} \lambda_i \geq 0 & \mu_i \geq 0 \\ y_i [1 - \xi_i - y_i(w^T x_i + b)] \leq 0 \\ \lambda_i [1 - \xi_i - y_i(w^T x_i + b)] = 0 \\ \xi_i \geq 0 & \mu_i \xi_i = 0 \end{cases}$$

若 $\lambda_i=0$ ，该样本不影响模型，否则 $1 - \xi_i - y_i(w^T x_i + b)=0$ ，即该样本为支持向量；由 $\lambda_i + \xi_i = C$ ，知当 $\lambda_i = C$ 时，则 $\mu_i = 0$ ，此时 $\xi_i \leq 1$ ，则该样本落在最大间隔内部，若则该样本被错误分类。由此可以看出，软间隔的最终模型仅与支持向量相关，保持了稀疏性。

软间隔最大化

$$w = \sum_{i=1}^m \lambda_i y_i x_i$$
$$b = \frac{1}{|S|} \sum_{s \in S} (y_s - w x_s)$$

然后通过上面两个式子求出 w 和 b ，最终求得超平面 $w^T x + b = 0$ ，

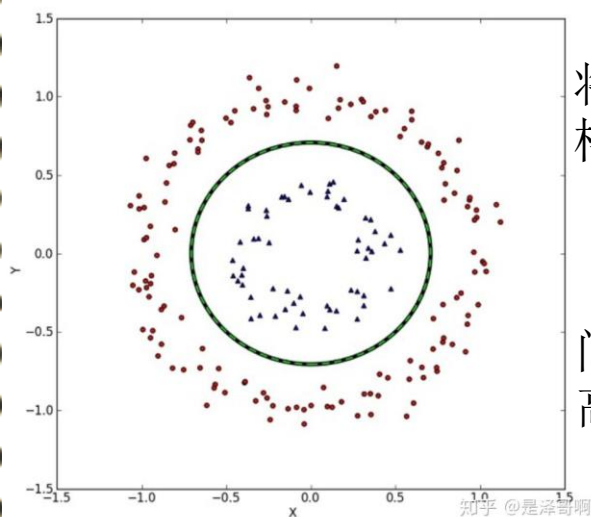
这边要注意一个问题，在间隔内的那部分样本点是不是支持向量？

我们可以由求参数 w 的那个式子可看出，只要 $\lambda_i > 0$ 的点都能够影响我们的超平面，因此都是支持向量。

核函数

核函数

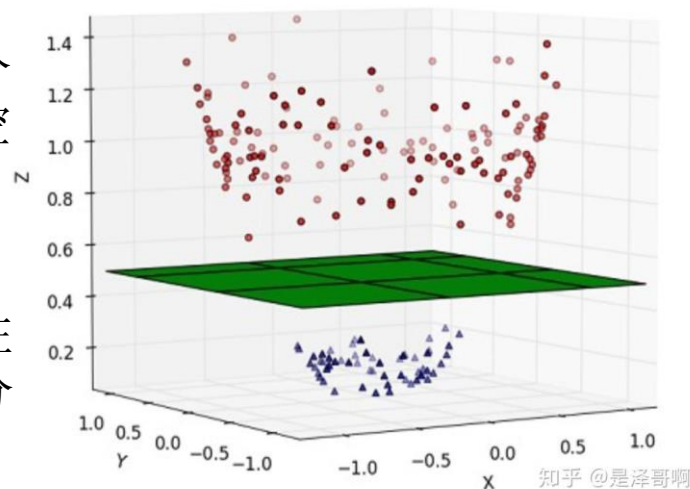
硬间隔和软间隔都是在说样本的完全线性可分或者大部分样本点的线性可分，无法处理样本点线性不可分的情况



将二维线性不可分
样本映射到高维空



间中，让样本点在
高维空间线性可分



知乎 @是泽哥啊

知乎 @是泽哥啊

核函数

若数据在原始的特征空间 R^d 不是线性可分的，支持向量机希望通过一个映射 $\phi: R^d \rightarrow R^{d^*}$ ，使得数据在新的空间 R^{d^*} 是线性可分的。

我们用 x 表示原来的样本点，用 $\phi(x)$ 表示 x 映射到特征新的特征空间 R^{d^*} 后到新向量。那么分割超平面可以表示为： $f(x) = w\phi(x) + b$ 。

对于非线性 SVM 的对偶问题就变成了：

$$\min_{\lambda} \left[\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j y_i y_j (\phi(x_i) \cdot \phi(x_j)) - \sum_{j=1}^n \lambda_j \right]$$
$$s. t. \quad \sum_{i=1}^n \lambda_i y_i = 0, \quad \lambda_i \geq 0, \quad C - \lambda_i - \mu_i = 0$$

注意到，在支持向量机的对偶型中，被映射到高维的特征向量总是以成对内积的形式存在，即 $\phi(x_i)^T \phi(x_j)$ 。如果先计算特征在空间 R^{d^*} 的映射，再计算内积，复杂度是 $O(d^*)$ 。当特征被映射到非常高维的空间，甚至是无穷维空间时，这将会是沉重的存储和计算负担。而更为重要的是我们往往不知道这个映射。

核函数

核技巧旨在将特征映射和内积这两步运算压缩为一步, 并且使复杂度由 $O(d^*)$ 降为 $O(d)$ 。即, 核技巧希望构造一个核函数 $\kappa(x_i, x_j)$, 使得 $\kappa(x_i, x_j) = \phi(x_i)^\top \phi(x_j)$, 并且 $\kappa(x_i, x_j)$ 的计算复杂度是 $O(d)$ 。

$$\begin{aligned} \min_{\lambda} & \left[\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j y_i y_j \boxed{k(x_i, x_j)} - \sum_{j=1}^n \lambda_j \right] \\ \text{s.t.} & \sum_{i=1}^n \lambda_i y_i = 0, \quad \lambda_i \geq 0, \quad C - \lambda_i - \mu_i = 0 \end{aligned} \quad \phi: x \mapsto \exp(-x^2) \begin{bmatrix} 1 \\ \sqrt{\frac{2}{1}}x \\ \sqrt{\frac{2^2}{2!}}x^2 \\ \vdots \end{bmatrix} \quad (41)$$

对应于核函数

$$\kappa(x_i, x_j) := \exp(-(x_i - x_j)^2). \quad (42)$$

证明.

$$\begin{aligned} \kappa(x_i, x_j) &= \exp(-(x_i - x_j)^2) \\ &= \exp(-x_i^2) \exp(-x_j^2) \exp(2x_i x_j) \\ &= \exp(-x_i^2) \exp(-x_j^2) \sum_{k=0}^{\infty} \frac{(2x_i x_j)^k}{k!} \\ &= \sum_{k=0}^{\infty} \left(\exp(-x_i^2) \sqrt{\frac{2^k}{k!}} x_i^k \right) \left(\exp(-x_j^2) \sqrt{\frac{2^k}{k!}} x_j^k \right) \\ &= \phi(x_i)^\top \phi(x_j). \end{aligned} \quad (43)$$

核函数的选择

用不同的核函数 $k(x, x_i)$ 可以构造实现输入空间中不同类型的非线性决策面的学习机，从而导致不同的支持向量算法。在实际问题中，通常直接给出核函数。常用的核函数有：

核函数名称	核函数表达式	核函数名称	核函数表达式
线性核	$\kappa(x, y) = x^T y$	指数核	$\kappa(x, y) = \exp\left(-\frac{\ x - y\ }{2\sigma^2}\right)$
多项式核	$\kappa(x, y) = (ax^T y + c)^d$	拉普拉斯核	$\kappa(x, y) = \exp\left(-\frac{\ x - y\ }{\sigma}\right)$
高斯核	$\kappa(x, y) = \exp\left(-\frac{\ x - y\ ^2}{2\sigma^2}\right)$	Sigmoid核	$\kappa(x, y) = \tanh(ax^T y + c)$

SVM分解算法

SVM分解算法

- SVM的计算问题

- 求解如下问题：

$$\max \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(x_i, x_j)$$

$$s.t. \quad 0 \leq \alpha_i \leq C, \quad \sum_i \alpha_i y_i = 0$$

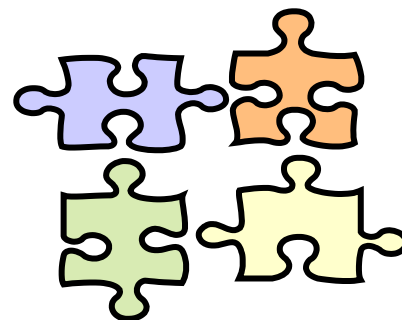
$$D = \{ d_{ij} | y_i y_j K(x_i, x_j) \}$$

D 的存储代价 = $0.5 \times (\text{训练样本数})^2 \times \text{单元存储空间}$

$$0.5 \times (7000)^2 \times 8 = 196 \text{ MByte}$$

SVM分解算法

- Edgar Osuna(Cambridge ,MA)等人在IEEE NNSP'97发表了An Improved Training Algorithm for Support Vector Machines ,提出了SVM的分解算法



SVM分解算法

- SVM的分解算法

- 将 α 向量分成两个集合,工作集 α_B , 固定集 α_N 。即

$$\alpha = \{\alpha_B, \alpha_N\}$$

- 每次对 α_B 解决一个小的二次规划问题, 保持 α_N 中的值不变
- 每次迭代选择不同的 α_B 和 α_N , 每解决一个小规模优化问题, 都在原来的基础上向最终的解集前进一步。
- 每次迭代检查当前结果, 满足优化条件, 则找到了优化问题的解, 算法结束。

SVM分解算法

- SVM的分解算法

- Original QP:

$$\max \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \varphi(x_i) \cdot \varphi(x_j)$$

$$0 \leq \alpha_i \leq C, \sum_i \alpha_i y_i = 0$$

- Small QP:

$$\max \quad \alpha^T \mathbf{1} - \frac{1}{2} \alpha^T \mathbf{D} \alpha$$

$$\mathbf{0} \leq \alpha \leq \mathbf{C}, \mathbf{y} = 0$$

$$\alpha = \begin{Bmatrix} \alpha_B \\ \alpha_N \end{Bmatrix}$$

$$\mathbf{D} = \begin{pmatrix} \mathbf{D}_{BB} & \mathbf{D}_{BN} \\ \mathbf{D}_{NB} & \mathbf{D}_{NN} \end{pmatrix}$$

SVM分解算法

- Small QP

$$\begin{aligned} \max \quad & \alpha_B^T \mathbf{1} - \frac{1}{2} [\alpha_B^T D_{BB} \alpha_B + \alpha_B^T D_{BN} \alpha_N + \alpha_N^T D_{NB} \alpha_B + \alpha_N^T D_{NN} \alpha_N] + \alpha_N \\ \text{s.t.} \quad & \alpha_B^T y_B + \alpha_N^T y_N = 0, \quad 0 \leq \alpha_B \leq C \end{aligned}$$

- 常数项: $-\frac{1}{2} \alpha_N^T D_{NN} \alpha_N + \alpha_N$

- 相等项: $\alpha_B^T D_{BN} \alpha_N = \alpha_N^T D_{NB} \alpha_B$

- 工作集B的规模人为指定

SVM分解算法

- 分解算法实现的关键步骤

- 任意从训练集选择 $|B|$ 个样本
- 求解 $|B|$ 个变量的二次优化子问题
- 当存在 α_j ($j \in N$), 使得

$$\alpha_j = 0 \quad \text{and} \quad g(x_j)y_j < 1$$

$$\alpha_j = C \quad \text{and} \quad g(x_j)y_j > 1$$

$$0 < \alpha_j < C \quad \text{and} \quad g(x_j)y_j \neq 1$$

- 用 α_j 替换工作集中的 α_i ($i \in B$)
- 重新求解子优化问题; 直到所有变量满足上述条件, 或者达到预定的循环次数

SVM分解算法

• 最小分解算法(SMO)

核心思想非常简单：每次只优化一个参数，其他参数先固定住，仅求当前这个优化参数的极值。

优化目标有约束条件： $\sum_i \alpha_i y_i = 0$ ，没法一次只变动一个参数。所以一次选择两个参数。具体步骤为：

1. 选择两个需要更新的参数 α_i 和 α_j ，固定其他参数。于是约束就变成了：

$$\alpha_i y_i + \alpha_j y_j = c \quad \alpha_i \geq 0, \alpha_j \geq 0$$

其中 $c = \sum_{k \neq i, j} \alpha_k y_k$ ，由此可以得出 $\alpha_j = (c - \alpha_i y_i) / y_j$ ，也就是说我们可以用 α_i 的表达式代替 α_j 。这样就相当于把目标问题转化成了仅有一个约束条件的最优化问题，仅有的约束是 $\alpha_i \geq 0$ 。

2. 对于仅有一个约束条件的最优化问题，我们完全可以在 α_i 上对优化目标求偏导，令导数为零，从而求出变量值 α_{i_new} ，然后根据 α_{i_new} 求出 α_{j_new} 。
3. 多次迭代直至收敛。

SVM分解算法

- SVM分解算法的实例

- SVM^{light}

Thorsten Joachims

(University Dortmund ,Informatik, AI-Unit)

<http://svmlight.joachims.org/>

<http://www.programsalon.com/>

参考文献: Make Large-Scale SVM Learning Practical

- SMO

John C. Platt

(Microsoft Research)

<http://theoval.sys.uea.ac.uk/svm/toolbox/>

参考文献: Fast Training of Support Vector Machines using Sequential Minimal Optimization

支持向量机

回归问题

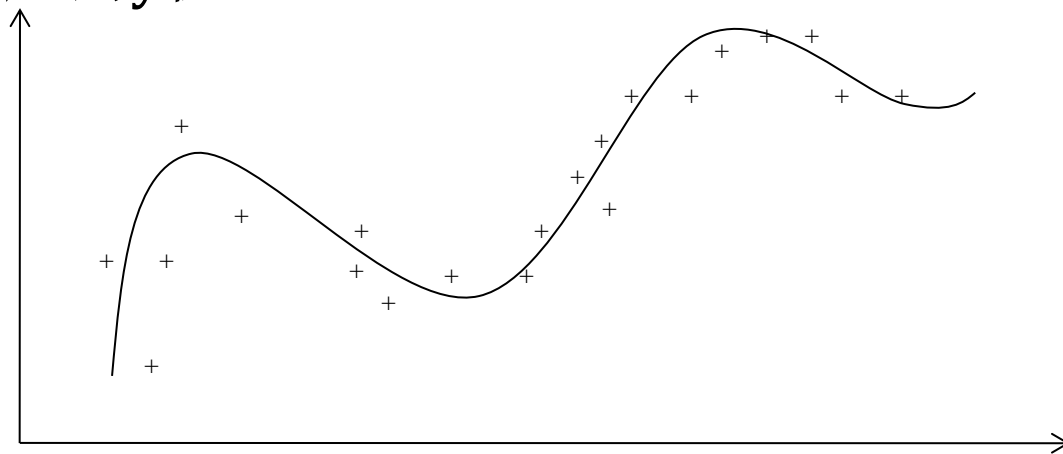
回归问题

- 回归问题描述

- 根据给定的训练集

$$T = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_l, y_l)\}$$

其中, $\mathbf{x}_i \in R^n, y_i \in R, i = 1, \dots, l$, 确定 R^n 上的一个实值函数 $f(\mathbf{x})$, 以便使用 $y = f(\mathbf{x})$ 来推断任意模式 \mathbf{x} 对应的 y 值。



$n=1$ 时函数回归的示意图

回归问题

• 回忆分类中的情形

- 已知: l 个观测样本, $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_l, y_l)$, $\mathbf{x}_i \in R^n, y_i \in \{-1, +1\}$
- 目标: 确定最优分类器 $y = f(\mathbf{x}, \mathbf{w})$
- 满足条件: 期望风险最小

$$R(\mathbf{w}) = \int L(y, f(\mathbf{x}, \mathbf{w})) dF(\mathbf{x}, y)$$

其中, $L(y, f(\mathbf{x}, \mathbf{w}))$ 称为损失函数, 通常为

$$L(y, f(\mathbf{x}, \mathbf{w})) = \begin{cases} 0 & y = f(\mathbf{x}, \mathbf{w}) \\ 1 & y \neq f(\mathbf{x}, \mathbf{w}) \end{cases}$$

回归问题

- SVM问题的数学表示

- SVM对应的优化问题（线性不可分情况）

- ✓ 已知： l 个观测样本 $(x_1, y_1), (x_2, y_2), \dots, (x_l, y_l)$

- ✓ 求解：

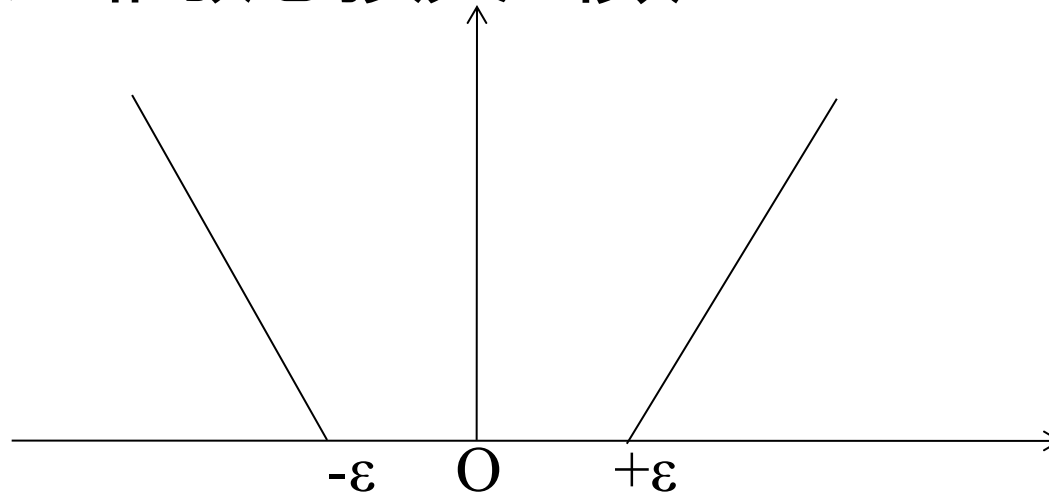
$$\begin{cases} \min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i \\ \text{s.t. } y_i(w \cdot \varphi(x_i) + b) \geq 1 - \xi_i, \\ \xi_i \geq 0 \ (i = 1, 2, \dots, l) \end{cases} \quad (P)$$

- 目标：确定最优分类超曲面

$$g(x) = w \cdot \varphi(x) + b = 0$$

回归问题

- 引入 ε -非敏感损失函数



ε -非敏感成本函数示意图

$$L(y, f(\mathbf{x}, \mathbf{w})) = |y - f(\mathbf{x})|_{\varepsilon} = \max\{0, |y - f(\mathbf{x})| - \varepsilon\}$$

$$= \begin{cases} |y - f(\mathbf{x})| - \varepsilon, & |y - f(\mathbf{x})| - \varepsilon > 0 \\ 0, & \text{otherwise} \end{cases}$$

回归问题

- SVM回归问题的数学表示

- SVM对应的优化问题

- ✓ 已知: l 个观测样本 $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_l, y_l)$

- ✓ 求解:
$$\begin{cases} \min \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*) \\ \text{s.t. } w \cdot \varphi(x_i) + b - y_i \leq \varepsilon + \xi_i, \\ y_i - w \cdot \varphi(x_i) - b \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \ (i = 1, 2, \dots, l) \end{cases} \quad (\text{P})$$

- 目标: 确定最优回归函数

$$f(\mathbf{x}) = w \cdot \varphi(\mathbf{x}) + b$$

回归问题

- 相应的Lagrange函数

$$\begin{aligned} L = & \frac{1}{2} ||w||^2 + C \sum_{i=1}^l (\xi_i + \xi_i^*) - \sum_{i=1}^l (\eta_i \xi_i + \eta_i^* \xi_i^*) \\ & - \sum_{i=1}^l \alpha_i (\varepsilon + \xi_i + y_i - w \cdot \varphi(x_i) - b) \\ & - \sum_{i=1}^l \alpha_i^* (\varepsilon + \xi_i^* - y_i + w \cdot \varphi(x_i) + b) \end{aligned}$$

- 其中, Lagrange乘子满足:

$$\alpha_i, \alpha_i^* \geq 0, \eta_i, \eta_i^* \geq 0$$

回归问题

- 最优解的条件为：

$$\nabla_b L = \sum_{i=1}^l (\alpha_i - \alpha_i^*) = 0$$

$$\nabla_w L = w - \sum_{i=1}^l (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) \varphi(x_i) \cdot \varphi(x_j) = 0$$

$$\nabla_{\xi} L = C - \alpha_i - \eta_i = 0$$

$$\nabla_{\xi^*} L = C - \alpha_i^* - \eta_i^* = 0$$

回归问题

- 原始问题的对偶问题

$$\begin{aligned} \max_{\alpha_i, \alpha_i^*} & -\frac{1}{2} \sum_{i=1}^l (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) \varphi(x_i) \cdot \varphi(x_j) \\ & - \sum_{i=1}^l y_i (\alpha_i^* - \alpha_i) - \varepsilon \sum_{i=1}^l (\alpha_i^* + \alpha_i) \\ \text{s. t.} & \sum_{i=1}^l (\alpha_i^* - \alpha_i) = 0, \\ & 0 \leq \alpha_i, \alpha_i^* \leq C, (i = 1, 2, \dots, l) \end{aligned}$$

回归问题

- 对偶问题引入核函数，即

$$\begin{aligned} \max_{\alpha_i, \alpha_i^*} & -\frac{1}{2} \sum_{i=1}^l (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j) K(x_i, x_j) \\ & - \sum_{i=1}^l y_i (\alpha_i^* - \alpha_i) - \varepsilon \sum_{i=1}^l (\alpha_i^* + \alpha_i) \\ \text{s.t.} & \sum_{i=1}^l (\alpha_i^* - \alpha_i) = 0, 0 \leq \alpha_i, \alpha_i^* \leq C, (i = 1, 2, \dots, l) \end{aligned}$$

回归问题

- SVM回归器的决策函数

$$f(x) = \sum_{i=1}^l (\alpha_i^* - \alpha_i) K(x_i, x) + b$$

- **注：**事实上，大部分拉格朗日乘子都为0，只有少数样本对应的拉格朗日乘子非0，并满足：

$$\alpha_i \alpha_i^* = 0$$

回归问题

- SVM回归器支持向量的稀疏性

- 稀疏性定理:

设 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_l)^T$, $\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_l^*)^T$

是SVM的解, 则有:

I. $\alpha_i, \alpha_i^* \in [0, C] \quad (i=1, 2, \dots, l);$

II. $\alpha_i \alpha_i^* = 0 \quad (i=1, 2, \dots, l)。$

回归问题

- 支持向量分布与拉格朗日乘子的关系

- 分布定理:

- 设 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_l)^T$, $\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_l^*)^T$

- 是SVM的解, 则有:

- I. 若 $\alpha_i = \alpha_i^* = 0$, 则相应的样本点一定位于 ε -带的内部或者边界上;
 - II. 若 $\alpha_i \in (0, C), \alpha_i^* = 0$ 或者 $\alpha_i = 0, \alpha_i^* \in (0, C)$, 则相应的样本点一定位于 ε -带的边界上;
 - III. 若 $\alpha_i = C, \alpha_i^* = 0$ 或者 $\alpha_i = 0, \alpha_i^* = 0$, 则相应的样本点一定位于 ε -带的外部或者边界上。

SVM方法的特点

- ① 对特征空间划分的最优超平面是SVM的目标,最大化分类边际的思想是SVM方法的核心;
- ② 非线性映射是SVM方法的理论基础,SVM利用内积核函数代替向高维空间的非线性映射;
- ③ 支持向量是SVM的训练结果,在SVM分类决策中起决定作用的是支持向量。

SVM 是一种有坚实理论基础的新颖的小样本学习方法。它基本上不涉及概率测度及大数定律等,因此不同于现有的统计方法。从本质上看,它避开了从归纳到演绎的传统过程,实现了高效的从训练样本到预报样本的“转导推理”(transductive inference),大大简化了通常的分类和回归等问题。

谢谢

附件：阅读材料

阅读材料

- An Overview of Statistical Learning Theory

http://www.mit.edu/~6.454/www_spring_2001/emin/slt.pdf

阅读材料

- SVM是一种基于统计学习理论的模式识别方法。
- 它是由Boser, Guyon, Vapnik在COLT-92上首次提出，从此迅速的发展起来，现在已经在许多领域（生物信息学，文本和手写识别等）都取得了成功的应用。

COLT(Computational Learning Theory)

阅读材料

- 经验风险的计算

- 期望风险 $R(w)$ 要依赖联合概率 $F(x, y)$ 的信息，实际问题中无法计算。
- 一般用经验风险 $R_{emp}(w)$ 代替期望风险 $R(w)$

$$R_{emp}(w) = \frac{1}{n} \sum_{i=1}^n L(y_i, f(x_i, w))$$



阅读材料

- 经验风险最小不等于期望风险最小，不能保证分类器的推广能力。
- 经验风险只有在样本数无穷大趋近于期望风险，需要非常多的样本才能保证分类器的性能。
- 需要找到经验风险最小和推广能力最大的平衡点。

阅读材料

- SVM的优势

- 传统机器学习的基础：数理统计理论

- ✓不足：过拟合，例如，神经网络

- SVM的基础：统计学习理论

- ✓优势1：不需要样本数量趋于无穷大

- ✓优势2：能够衡量学习函数集的复杂程度

- ✓优势3：把传统期望风险的估计拓展成结构风险