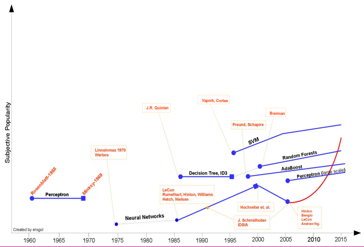
支持向量机

主要内容

SVM概述
最大分类间隔
支持向量机
二次凸优化
对偶学习算法

SVM

•支持向量机(support vector machines. SVM)




Vladimir Vapnik

SVM

•支持向量机(support vector machines. SVM)

•二类分类模型.它的基本模型是定义在特征空间上的间隔最大的线性分类器，间隔最大使它有别于感知机；

•支持向量机还包括核技巧，这使它成为实质上的非线性分类器。

•支持向量机的学习策略就是间隔最大化，可形式化为一个求解凸二次规划(convex quadratic programming)的问题，也等价于正则化的合页损失函数的最小化问题.支持向量机的学习算法是求解凸二次规划的最优化算法。

SVM分类

•支持向量机(supportvectormachines. SVM)

•线性可分支持向量机(linear support vector machine in linearly separable case).
- 硬间隔最大化(hard margin maximization)；

•线性支持向量机(linear supportvector machine)
- 训练数据近似线性可分时，通过软间隔最大化(soft margin maximization)；

•非线性支持向量机(non-linear support vector machine)
- 当训练数据线性不可分时，通过使用核技巧(kernel trick)及软间隔最大化。

线性可分问题

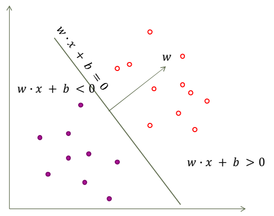
•假设特征空间上的训练数据集：

$$T=\{(x_1,y_1),(x_2,y_2),\dots,(x_N,y_N)\}$$
$$x_i\in\mathcal{X}=R^n, y_i\in\mathcal{Y}=\{+1,-1\}, i=1,2,\dots,N$$

•正例和负例
•学习的目标：找到分类超平面，
•回顾：感知机模型，学习得到的分离超平面
•决策函数：

$$w^*\cdot x+b^*=0$$
$$f(x)=sign(w^*\cdot x+b^*)$$

线性可分支持向量机与硬间隔最大化



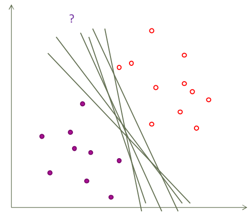
超平面选择

经验风险最小
- 最小化训练集错分
- 解不唯一
- 容易受数据集缺陷影响

那是否可以从众多的分类超平面选择一个最优的？

需要新的标准

结构风险最小

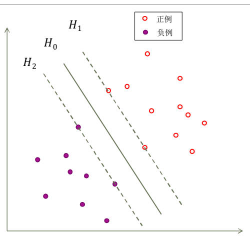


超平面选择

经验风险最小
- 最小化训练集错分
- 解不唯一
- 容易受数据集缺陷影响

思考：
是否可以从众多的分类超平面选择一个最优的？

需要新的标准
- 结构风险最小
- 最大分类间隔



在某个分类超平面H_0的基础上再寻找两个超平面H_1和H_2

H_0:　$w \cdot x + b = 0$

H_1:　$w \cdot x + b = r$

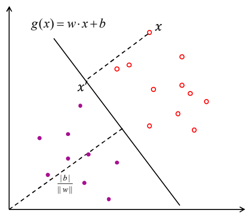
x_i为正例:　$w \cdot x + b \geq r$

H_2:　$w \cdot x + b = -r$

x_i为负例:　$w \cdot x + b \leq -r$

等间距、平行

点到超平面的距离



$|w \cdot x + b|$

- 点到分离超平面的远近
- 表示分类预测的确信程度

$y(w \cdot x + b)$

- 分类的正确性
- 分类的确信度

函数间隔和几何间隔

- 函数间隔
  - 样本点的函数间隔

$$\hat{\gamma}_i = y_i(w \cdot x_i + b)$$

  - 训练数据集的函数间隔

$$\hat{\gamma} = \min_{i=1,\dots,N} \hat{\gamma}_i$$

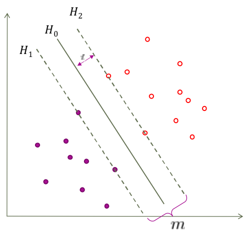
  - 表示分类预测的正确性和确信度
- 超平面不变的情况下
  - 函数间隔随w和b的变化而成比例的改变
- 为了使间隔是确定的
  - 对法向量w进行规范化
  - $||w|| = 1$
- 几何间隔
  - 样本的几何间隔

$$\gamma_i = y_i\left(\frac{w}{\|w\|} \cdot x_i + \frac{b}{\|w\|}\right) \qquad \gamma_i = \frac{\hat{\gamma}_i}{\|w\|}$$

  - 数据集的几何间隔

$$\gamma = \min_{i=1,\dots,N} \gamma_i \qquad \gamma = \frac{\hat{\gamma}}{\|w\|}$$

最大分类间隔



H_0:　$w \cdot x + b = 0$

H_1:　$w \cdot x + b = r$

H_2:　$w \cdot x + b = -r$

$$d(H_1, H_0) = \frac{|(b-r)-b|}{||w||}$$

$$d = \frac{r}{||w||}$$

$$m = 2d = \frac{2r}{||w||}$$

令 $r = 1$

间隔 $m = \frac{2}{||w||}$

间隔最大化

- 最大间隔分类超平面

$$\max_{w,b} \frac{2}{||w||}$$
$$s.t. \quad y_i(\frac{w}{\|w\|} \cdot x_i + \frac{b}{\|w\|}) \geq 1, i = 1, 2, ..., N$$

- 最大化 $\frac{2}{\|w\|}$ 等价于最小化 $\frac{1}{2}\|w\|^2$

线性可分支持向量机

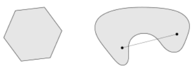
- 线性可分支持向量机学习的最优化问题

$$\min_{w,b} \frac{1}{2}\|w\|^2$$
$$s.t. \quad y_i(w \cdot x_i + b) - 1 \geq 0, i = 1, 2, ..., N$$

- 凸二次规划(convex quadratic programming)

凸集

- 一个点集（或区域），如果连接其中任意两点x_1、x_2的线段都全部包含在该集合内，就称该点集为凸集，否则为非凸集。

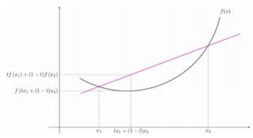


凸函数

如果一个函数满足以下条件
（1）它的定义域dom(f)是R^n上的凸集，并且
（2）对于所有的x_1，x_2 ∈dom(f)且$\alpha \in (0,1)$

$$f(\alpha x_1 + (1-\alpha)x_2) \leq \alpha f(x_1) + (1-\alpha) f(x_2)$$

那么函数f为凸函数。

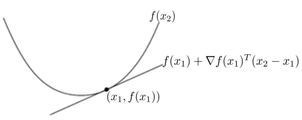


凸性条件

1.根据一阶导数（函数的梯度）来判断函数的凸性设$f(x)$为定义在凸集R上，且具有连续的一阶导数的函数，则$f(x)$在R上为凸函数的充要条件是对凸集R内任意不同两点x_1 与x_2 ，不等式

$$f(x_2) \geq f(x_1) + (x_2 - x_1)^T \nabla f(x_1)$$

恒成立。



凸性条件

2.根据二阶导数(Hessian矩阵)来判断函数的凸性设$f(x)$为定义在凸集R上且具有连续二阶导数的函数，则$f(x)$在R上为凸函数的充要条件：

Hessian矩阵在R上处处半正定

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \succeq 0$$

凸优化问题

- 对于约束优化问题

$$\min f(x)$$
$$s.t.\quad g_j(x) \le 0 \quad j=1,2,...,m$$
$$h_i(w)=0, i=1,2,...,l$$

- 此问题为凸规划：
  - 若 $f(x)$与 $g_j(x)$都为连续可微的凸函数。
  - 若 $h_j(x)$ 为仿射变换。
- 凸规划的任何局部最优解就是全局最优解。
- 当目标函数为二次函数，g函数为仿射函数时，为凸二次规划问题。

线性可分支持向量机学习算法

- 输入：线性可分训练数据集$T=\{(x_1,y_1),(x_2,y_2),...,(x_N,y_N)\}$

$$x_i \in \mathcal{X} = R^n, y_i \in \mathcal{Y} = \{+1,-1\}, i=1,2,...,N$$

- 输出：最大间隔分离超平面和分类决策函数

1、构造并求解约束最优化问题

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2$$
$$s.t.\quad y_i(w\cdot x_i+b)-1 \ge 0, i=1,2,...,N$$

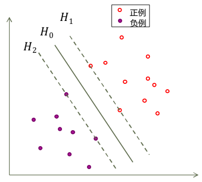
求得w^*和b^*

2、得到分离超平面 $w^*\cdot x+b^*=0$

分类决策函数$f(x)=sign(w^*\cdot x+b^*)$

支持向量和Margins（边界）

- 在线性可分情况下，训练数据集的样本点中与分离超平面距离最近的样本点的实例称为支持向量(support vector)；
- 支持向量是使约束条件式等号成立的点，即$y_i(w\cdot x_i+b)-1=0$
- 正例：$H_1: w\cdot x+b=1$
- 负例：$H_2: w\cdot x+b=-1$



拉格朗日对偶

常规二次规划算法

- 约束数量多
- 优化问题复杂

拉格朗日对偶算法

- 更容易求解
- 方便使用核函数

拉格朗日对偶

- 原始问题：

$$\min_{w,b} \quad \frac{1}{2}\|w\|^2$$
$$s.t.\quad y_i(w\cdot x_i+b)-1 \ge 0, i=1,2,...,N$$

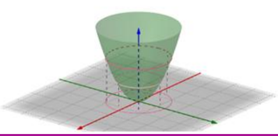
- 在约束最优化问题中，常常利用拉格朗日对偶性(Lagrange duality)将原始问题转换为对偶问题，通过解对偶问题得到原始问题的解。

拉格朗日对偶

1、原始问题

- 设$f(x),c(x),h(x)$是定义在R^n上的连续可微函数

$$\min_{x\in R^n} \quad f(x)$$
$$s.t.\quad c_i(x)\le 0, i=1,2...,k$$
$$h_j(x)=0, j=1,2...,l$$

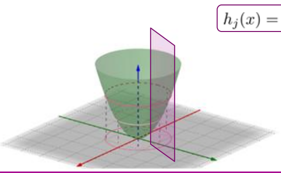


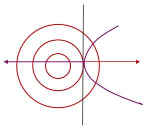
拉格朗日对偶

1、原始问题

- 设$f(x), c(x), h(x)$是定义在R^n上的连续可微函数

$$\min_{x\in R^n} f(x)$$
$$s.t.\quad c_i(x)\le 0, i=1,2\ldots,k$$
$$\boxed{h_j(x)=0, j=1,2\ldots,l}$$





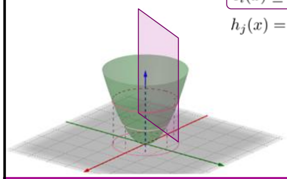
$$\nabla f+\alpha\nabla h_j=0$$
$$L_j(x,\alpha_j)=f(x)+\alpha_j h_j(x)$$

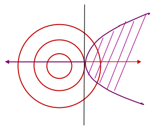
拉格朗日对偶

1、原始问题

- 设$f(x), c(x), h(x)$是定义在R^n上的连续可微函数

$$\min_{x\in R^n} f(x)$$
$$s.t.\quad \boxed{c_i(x)\le 0, i=1,2\ldots,k}$$
$$h_j(x)=0, j=1,2\ldots,l$$





$$\nabla f+\alpha_i\nabla c_i=0\quad \alpha_i\ge 0$$
$$L_i(x,\alpha_i)=f(x)+\alpha_i c_i(x)$$

拉格朗日对偶

1、原始问题

- 设$f(x), c(x), h(x)$是定义在R^n上的连续可微函数

$$\min_{x\in R^n} f(x)$$
$$s.t.\quad c_i(x)\le 0, i=1,2\ldots,k$$
$$h_j(x)=0, j=1,2\ldots,l$$

- 引进拉格朗日函数α_i,β_j 为乘子$\alpha_j\ge 0$

$$L(x,\alpha,\beta)=f(x)+\sum_{i=1}^{k}\alpha_i c_i(x)+\sum_{j=1}^{l}\beta_j h_j(x)$$

拉格朗日对偶

$$L(x,\alpha,\beta)=f(x)+\sum_{i=1}^{k}\alpha_i c_i(x)+\sum_{i=1}^{l}\beta_j h_j(x)$$

- 考虑x的函数，P为原始问题 $\theta_P(x)=\max_{\alpha,\beta:\alpha_i\ge 0} L(x,\alpha,\beta)$
- 假设给定某个x，如果x违反约束条件：

$$c_i(x)>0\qquad h_j(x)\ne 0$$
$$\theta_P(x)=\max_{\alpha,\beta:\alpha_i\ge 0}[f(x)+\sum_{i=1}^{k}\alpha_i c_i(x)+\sum_{i=1}^{l}\beta_j h_j(x)]=+\infty$$

拉格朗日对偶

$$L(x,\alpha,\beta)=f(x)+\sum_{i=1}^{k}\alpha_i c_i(x)+\sum_{i=1}^{l}\beta_j h_j(x)$$

- 考虑优化问题 $\theta_P(x)=\max_{\alpha,\beta:\alpha_i\ge 0} L(x,\alpha,\beta)$
- 假设给定某个x，如果x符合约束条件：

$h_j(w)=0$ $\quad\nabla_x L=0$ 时取最优

$c_i(w)=0$ $\quad\nabla_x L=0$ 时取最优 且 $\alpha>0$

$c_i(w)<0$ $\quad\nabla_x L=0$ 时取最优 且 $\alpha=0$

$\Rightarrow \alpha_i c_i(x)=0, \alpha\ge 0$

$$\theta_P(x)=\max_{\alpha,\beta:\alpha_i\ge 0}[f(x)+\sum_{i=1}^{k}\alpha_i c_i(x)+\sum_{i=1}^{l}\beta_j h_j(x)]=f(x)$$

拉格朗日对偶

$$\theta_P(x)=\begin{cases} f(x), & x\ \text{满足原始问题约束} \\ +\infty, & \text{其他} \end{cases}$$

- 考虑极小问题：

$$\min_x \theta_P(x)=\min_x \max_{\alpha,\beta:\alpha_i\ge 0} L(x,\alpha,\beta)$$

- 与原始最优化问题等价

$$p^*=\min_x \theta_P(x)$$

拉格朗日对偶

2、对偶问题

定义：$\theta_D(\alpha,\beta)=\min_x L(x,\alpha,\beta)$

$$\max_{\alpha,\beta:\alpha_i\geq 0}\theta_D(\alpha,\beta)=\max_{\alpha,\beta:\alpha_i\geq 0}\min_x L(x,\alpha,\beta)$$

则最大值问题：

- 称为广义拉格朗日函数的极大极小问题
- 表示为约束最优化问题：$\max_{\alpha,\beta}\theta_D(\alpha,\beta)=\max_{\alpha,\beta}\min_x L(x,\alpha,\beta)$ $s.t.\quad \alpha_i\geq 0, i=1,2,...,k$
- 称为原始问题的对偶问题，
- 对偶问题的最优值 $d^*=\max_{\alpha,\beta:\alpha_i\geq 0}\theta_D(\alpha,\beta)$

原始问题和对偶问题的关系

- **定理：**
- 若原始问题和对偶问题都有最优值，则：

$$d^*=\max_{\alpha,\beta:\alpha_i\geq 0}\min_x L(x,\alpha,\beta)\leq\min_x\max_{\alpha,\beta:\alpha_i\geq 0}L(x,\alpha,\beta)=p^*$$

- 推论：
- 设x*，和α*，β*分别是原始问题和对偶问题的可行解，并且$d^*=$ p*，则x^*，和α*，β*分别是原始问题和对偶问题的最优解

KKT条件

- **定理：**对原始问题和对偶问题，假设函数$f(x)$和$c_i(x)$是凸函数，$h_j(x)$是仿射函数，并且不等式$c_i(x)$是严格可行的，则x^*，和α*，β*分别是原始问题和对偶问题的解的充分必要条件是x^*，和α*，β*满足karush-Kuhn-Tucker(KKT)条件。

$$\nabla_x L(x^*,\alpha^*,\beta^*)=0$$

$$\alpha_i^*c_i(x^*)=0, i=1,2,...,k \quad\Rightarrow\quad \text{KKT互补条件}$$

$$c_i(x^*)\leq 0, i=1,2,...,k$$

$$\alpha_i^*\geq 0, i=1,2,...,k$$

$$h_j(x^*)=0, j=1,2,...,l$$

SVM的对偶算法

- 对于线性可分支持向量机的优化问题，原始问题：

$$\min_{w,b}\ \frac{1}{2}\|w\|^2$$
$$s.t.\quad y_i(w\cdot x_i+b)-1\geq 0, i=1,2,...,N$$

- 应用拉格朗日对偶性，通过求解对偶问题，得到原始问题的解。
- 优点：
  - 对偶问题往往容易易解
  - 引入核函数，推广到非线性分类问题

SVM的对偶算法

- 定义拉格朗日函数

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

- 原问题：极小极大，对偶问题：极大极小

$$\min_{w,b}\theta_P(w,b)=\min_{w,b}\max_{\alpha:\alpha_i\geq 0}L(w,b,\alpha)$$

$$\Downarrow$$

$$\max_{\alpha:\alpha\geq 0}\min_{w,b}L(w,b,\alpha)$$

学习的对偶算法

- 先求$L(w,b,\alpha)$对w，b的极小，再求对α的极大
- 1、求：$\min_{w,b}L(w,b,\alpha)$ ，对w，b分别求偏导并令等于0

由 $\nabla_w L(w,b,\alpha)=w-\sum_{i=1}^N\alpha_i y_i x_i=0$ $\Rightarrow$ $w=\sum_{i=1}^N\alpha_i y_i x_i$

$\nabla_b L(w,b,\alpha)=\sum_{i=1}^N\alpha_i y_i=0$ $\qquad$ $\sum_{i=1}^N\alpha_i y_i=0$

得：

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

$$\Updownarrow$$

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

SVM的对偶算法

- 求 $\min_{w,b} L(w,b,\alpha)$ 对α的极大：

$$\max_{\alpha} \quad -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i\cdot x_j)+\sum_{i=1}^{N}\alpha_i$$
$$s.t. \quad \sum_{i=1}^{N}\alpha_i y_i = 0$$
$$\alpha_i \geq 0,\ i=1,2,...,N$$

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i\cdot x_j)-\sum_{i=1}^{N}\alpha_i$$
$$s.t. \quad \sum_{i=1}^{N}\alpha_i y_i = 0$$
$$\alpha_i \geq 0,\ i=1,2,...,N$$

SVM的对偶算法

- 定理：设$\alpha^* = (\alpha_1^*, \alpha_2^*, ..., \alpha_l^*)^T$ 是对偶最优问题的解，
  则存在下标 j，使得$\alpha_j^* > 0$，并可按下式求得原始问题的解。

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

证明：由

$$\nabla_w L(w^*, b^*, \alpha^*) = w^* - \sum_{i=1}^{N}\alpha_i^* y_i x_i = 0$$
$$\nabla_b L(w^*, b^*, \alpha^*) = -\sum_{i=1}^{N}\alpha_i^* y_i = 0$$
$$\alpha_i^*(y_i(w^*\cdot x_i + b^*) - 1) = 0, i = 1,2,...,N$$
$$y_i(w^*\cdot x_i + b^*) - 1 \geq 0, i = 1,2,...,N$$
$$\alpha_i^* \geq 0, i = 1,2,...,N$$

得： $w^* = \sum_i \alpha_i^* y_i x_i$

SVM的对偶算法

- 定理：设$\alpha^* = (\alpha_1^*, \alpha_2^*, ..., \alpha_l^*)^T$ 是对偶最优问题的解
- 则存在下标 j，使得$\alpha_j^* > 0$ ，并可按下式求得原始问题的解。
- 证明：由$w^* = \sum_i \alpha_i^* y_i x_i$ ，其中至少有一个$\alpha_j^* > 0$

  反证法：

  假设：$\alpha^* = 0$ ，则$w*$=0，

  但这不是原始优化问题的解，产生矛盾

  对此：j 有$y_j(w^*\cdot x_j + b^*) - 1 = 0$

  将式$w^* = \sum_{i=1}^{N}\alpha_i^* y_i x_i$ 代入并注意到$y_j^2 = 1$

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

SVM的对偶算法

- 由此定理可知，分离超平面可以写成：

$$\sum_{i=1}^{N}\alpha_i^* y_i (x\cdot x_i) + b^* = 0$$

- 分类决策函数可以写成：

$$f(x) = sign(\sum_{i=1}^{N}\alpha_i^* y_i (x\cdot x_i) + b^* = 0)$$

- 这就是说，分类决策函数只依赖于输入x和训练样本输入的内积，上式称为线性可分支持向量机的对偶形式。

线性可分支持向量机学习算法

- 输入：线性可分训练数据集$T = \{(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)\}$

$$x_i \in \mathcal{X} = R^n, y_i \in \mathcal{Y} = \{+1, -1\}, i = 1,2,...,N$$

- 输出：最大间隔分离超平面和分类决策函数
- 1、构造并求解约束最优化问题

$$\min_{\alpha} \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j (x_i\cdot x_j)-\sum_{i=1}^{N}\alpha_i$$
$$s.t. \quad \sum_{i=1}^{N}\alpha_i y_i = 0$$
$$\alpha_i \geq 0,\ i=1,2,...,N$$

- 求得最优解：

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

线性可分支持向量机学习算法

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

并选择α^*的一个正分量$\alpha_j^* > 0$，计算

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

3、求得分离超平面 $w^*\cdot x + b^* = 0$

分类决策函数 $f(x) = sign(w^*\cdot x + b^*)$

支持向量

- 考虑原始优化问题和对偶优化问题，
- 将数据集中对应于$\alpha_j^* > 0$ 的样本(x_i, y_i) 的实例$x_i \in R^n$
- 称为支持向量

- 支持向量一定在分割边界上，由KKT互补条件：

$$\alpha_i^*(y_i(w^* \cdot x_i + b^*) - 1) = 0, i = 1, 2, ..., N$$

- 对应于$\alpha_j^* > 0$ 的样本 x_i

$$y_i(w^* \cdot x_i + b^*) - 1 = 0$$

- 或

$$w^* \cdot x_i + b^* = \pm 1$$

结构化风险与正则化

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