

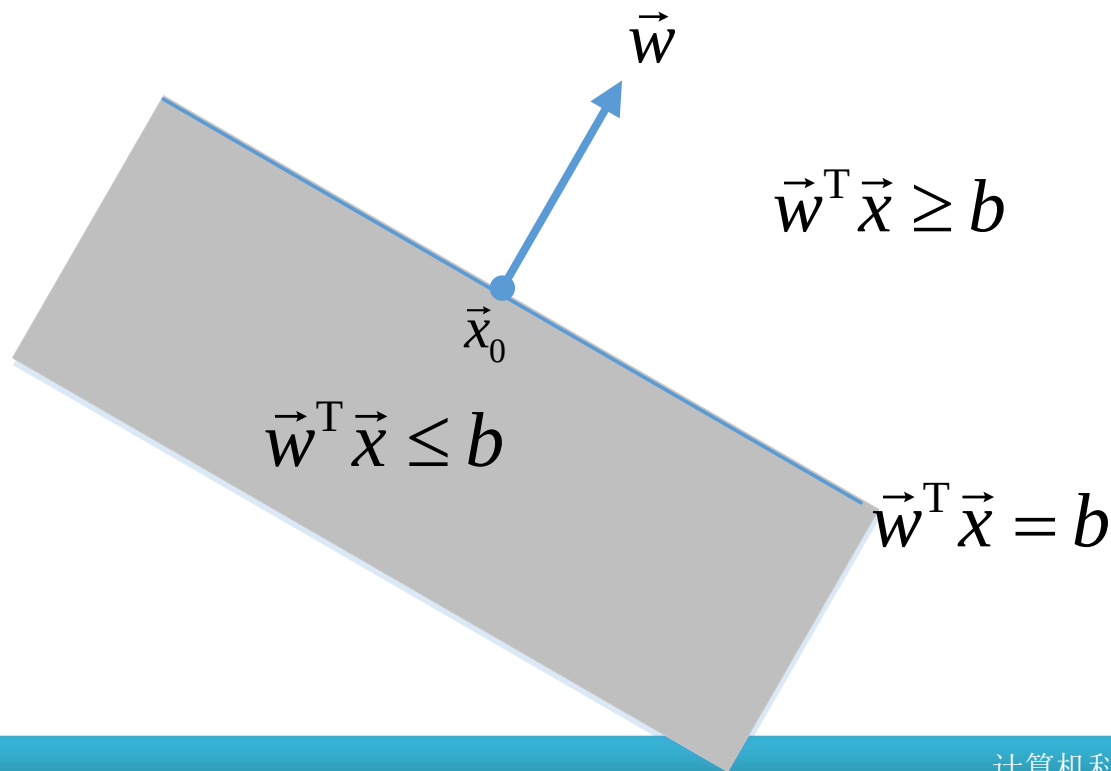
基于向量空间模型的文本分类

内容

- 二元线性SVM
- SVM用于非线性分类

用超平面来分割多维空间

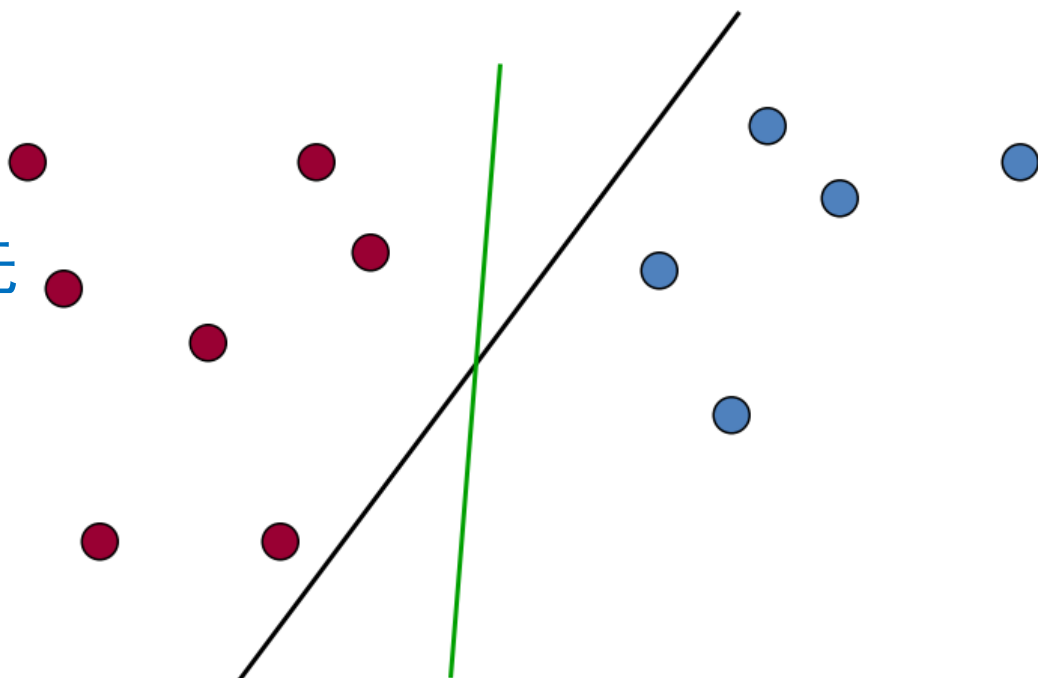
- A hyperplane is a set of the form $\{\vec{x} \mid \vec{w}^T \vec{x} = b\}$, where $\vec{w} \in \mathbb{R}^n$, $\vec{w} \neq 0$, and $b \in \mathbb{R}$.
- A hyperplane divides $\mathbb{R}^n$ into two halfspaces. A (closed) halfspace is a set of the form $\{\vec{x} \mid \vec{w}^T \vec{x} \leq b\}$



应该选哪个超平面？

- 对于线性可分的训练集而言，肯定存在无穷多个分类面可以将两类完全正确地分开，但是不同的分类面在测试集的表现完全迥异...
- 对于新数据，有些分类器的错误率很高，有一些却很低。如感知机：通常很差；朴素贝叶斯、Rocchio：一般；线性SVM：好

对于二类线性可分问题存在无穷多个能区分两类的超平面
直观的感觉是绿色的更好

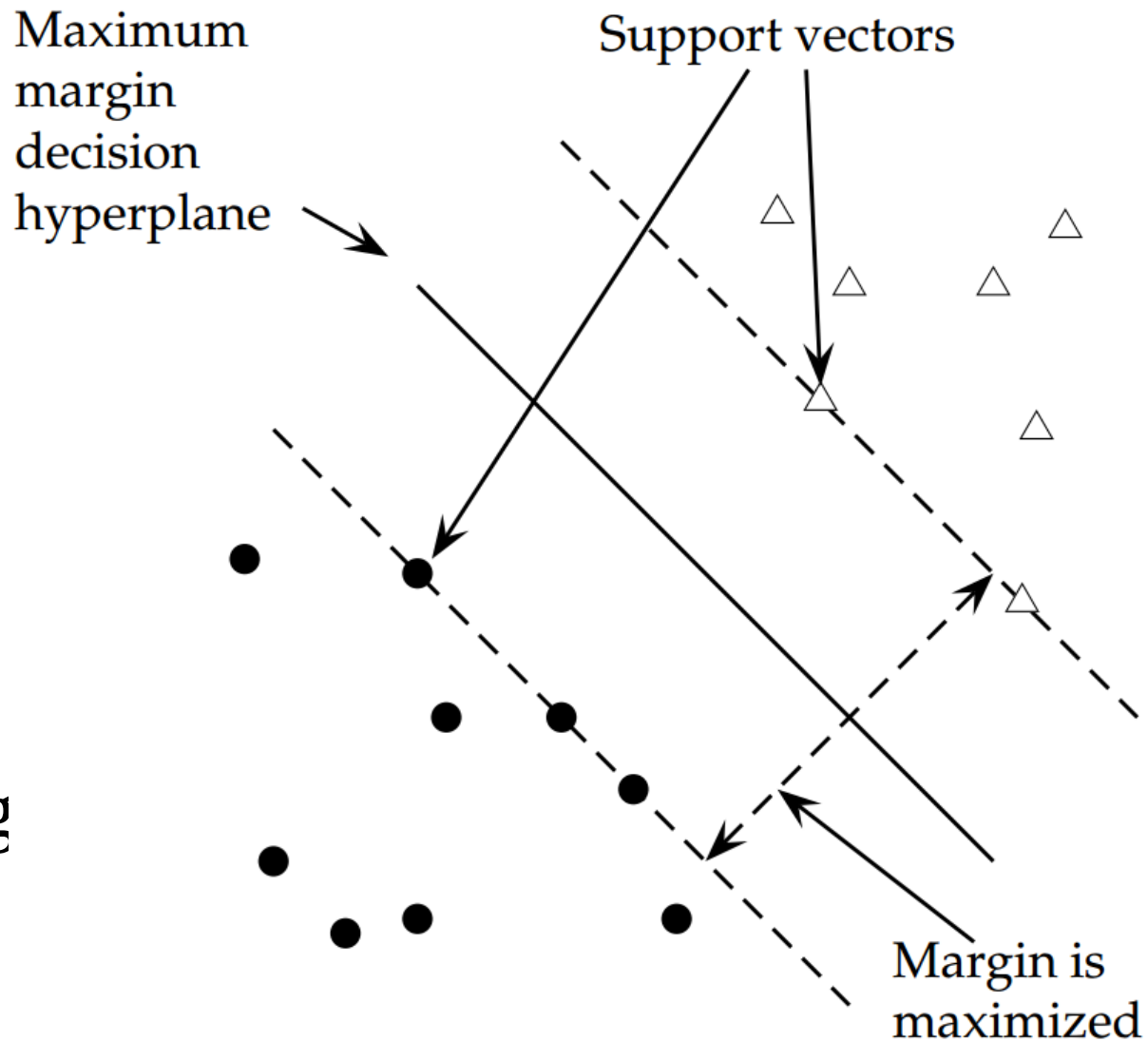


分类器的间隔

- 有些学习方法(如感知机)只需找到任一线性分界面即可，而另一些方法(如**NB**)则需按照某个准则找到最优的线性分界面。
 - **SVM**是最大间隔分类器的一种，是基于向量空间的机器学习方法，其目标是找到两个类别之间的一个决策边界，使之尽量远离训练集上的任意一点(当然一些离群点和噪音点可能不包括在内)
 - **SVM**定义的准则是寻找一个离数据点最远的决策面。从决策面到最近数据点的距离决定了分类器的间隔(**margin**)

支持向量机(SVM, Support Vector Machines)

- SVMs **maximize** the margin around the separating hyperplane.
- The decision function is fully specified by a subset of training samples, the **support vectors**.



一个SVM的例子：几何求解

- 两点(1, 1)和(2, 3)
- 两点确定直线： $x_2 = 2x_1 - 1$
- 最优的分类直线：与上述线段垂直并相交与其中点(1.5, 2)，既(中垂线)
- 于是，可以求得SVM的决策直线方程为：

$$x_2 = -\frac{1}{2}x_1 + \frac{11}{4}$$

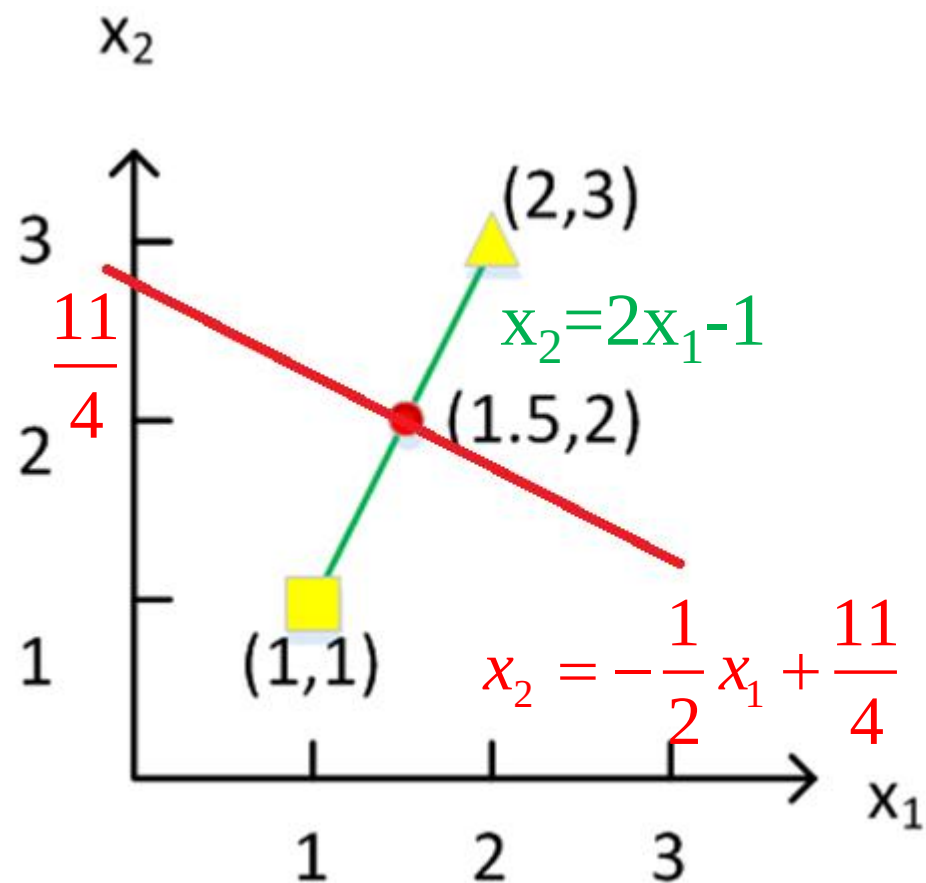
– $x_2 = ax_1 + b$

– 斜率(通过与绿线垂直):

$$a = -1/2$$

– 通过点(1.5, 2)

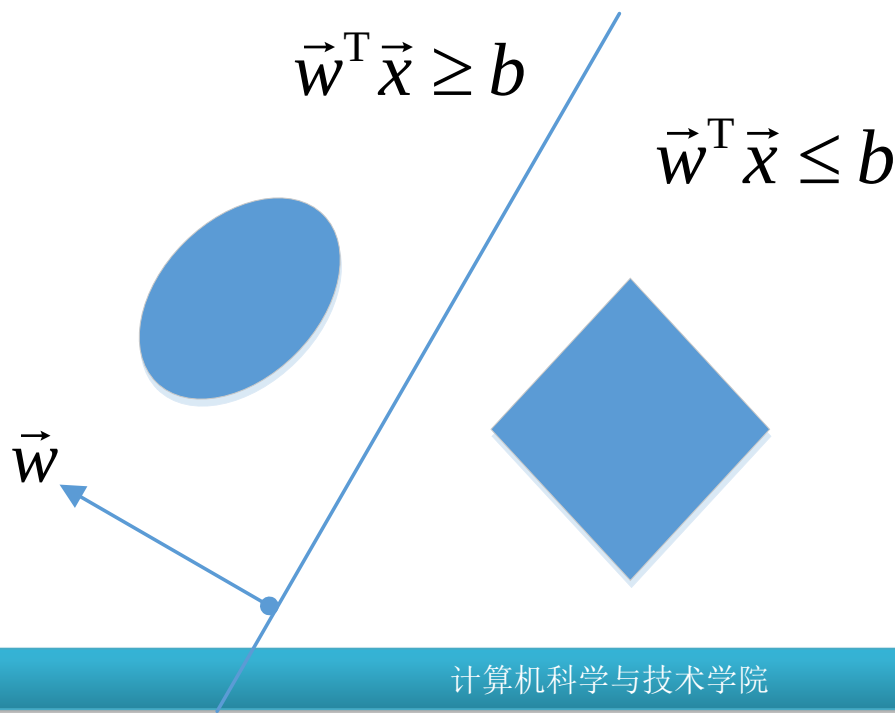
$$2 = -\frac{1}{2} \times 1.5 + b \Rightarrow b = \frac{11}{4}$$



二元线性分类器的形式化定义

- $\vec{w}$: decision hyperplane normal vector
- $\vec{x}_i$: data point i
- y_i : class of data point i (+1 or -1)
- Classifier is: $f(\vec{x}_i) = \text{sign}(\vec{w}^T \vec{x}_i + b)$

sign符号函数，取值+1或-1

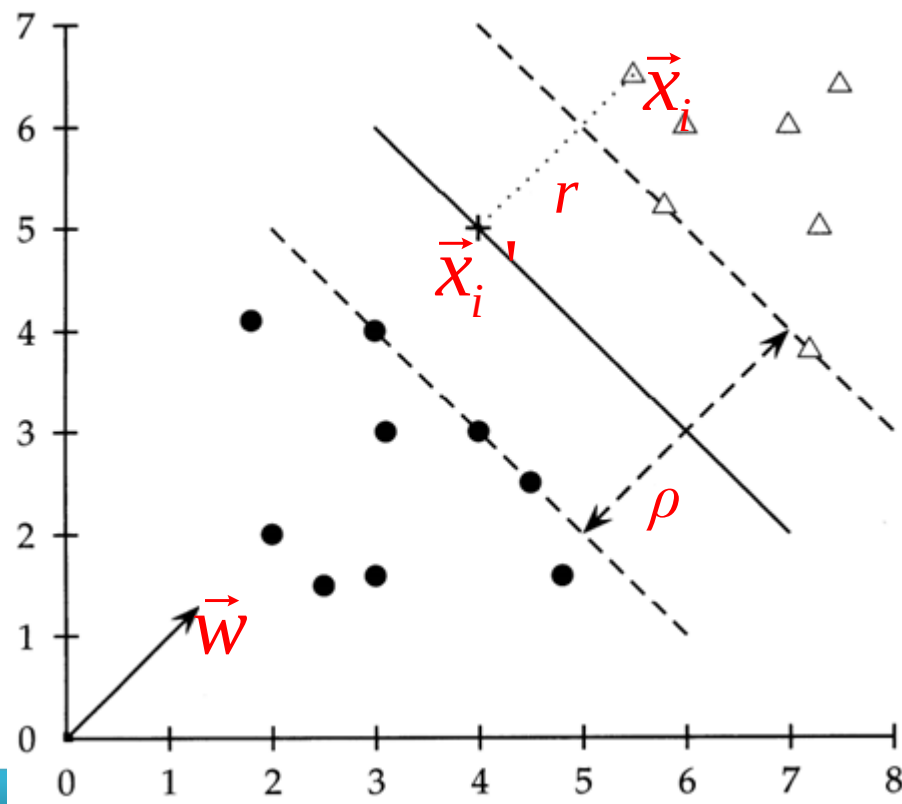


函数间隔 Functional Margin

- 点 $\vec{x}_i$ 相对于超平面 $\langle \vec{w}, b \rangle$ 的函数间隔

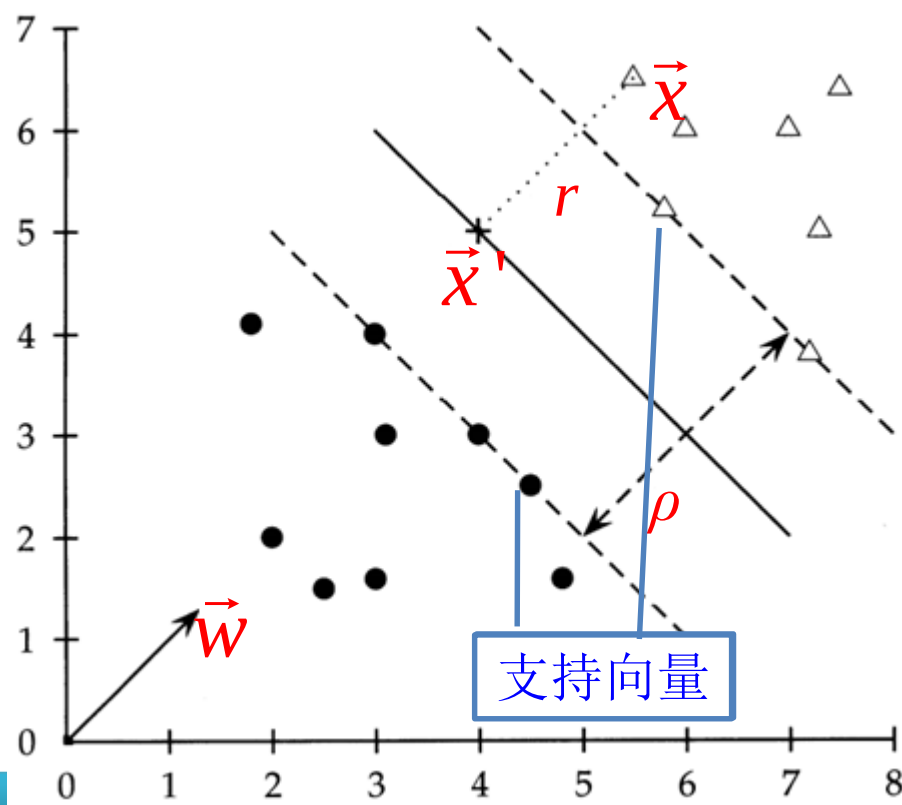
$$y(\vec{w}^T \vec{x}_i + b)$$

- 一个数据集的函数间隔是数据集点中最小函数间隔的两倍(超平面左右各一部分)
- 问题：具体的值没有大小限制，可以通过放大 $\vec{w}$ 和 b 来获得所需要的任意函数边界。
 - 例如，如果将 $\vec{w}$ 和 b 分别替换成 $5\vec{w}$ 和 $5b$ ，那么函数间隔就变成，
$$y_i(5\vec{w}^T \vec{x}_i + 5b)$$
即值变大了五倍



几何间隔Geometric Margin

- 是中间空白带的最大宽度，该空白带可以用于将两类支持向量分开，是 r 的最小值(即支持向量到超平面的距离)的2倍
- 点 $\vec{x}$ 到超平面的最短欧式距离 r : $r = \frac{y(\vec{w}^T \vec{x} + b)}{|\vec{w}|}$ 函数间隔
- 离超平面最近的点是支持向量(support vector)
- 不管参数如何缩放，几何间隔总是一个不变量(因为分母做了归一化处理)
- 几何间隔是实际的点到直线的距离



点 $\vec{x}$ 到超平面的几何间隔 r 的分析:

- 将超平面上离 $\vec{x}$ 最近的点标记为 $\vec{x}'$ ，那么线 $\vec{x} - \vec{x}'$ 与法向量 $\vec{w}$ 平行
- 法向量的单位向量是 $\vec{w}/|\vec{w}|$
- 因此 $\vec{x} - \vec{x}'$ 就是 $r * \vec{w}/|\vec{w}|$ 平移的结果
- 所以(y 表示点在上面或下面)

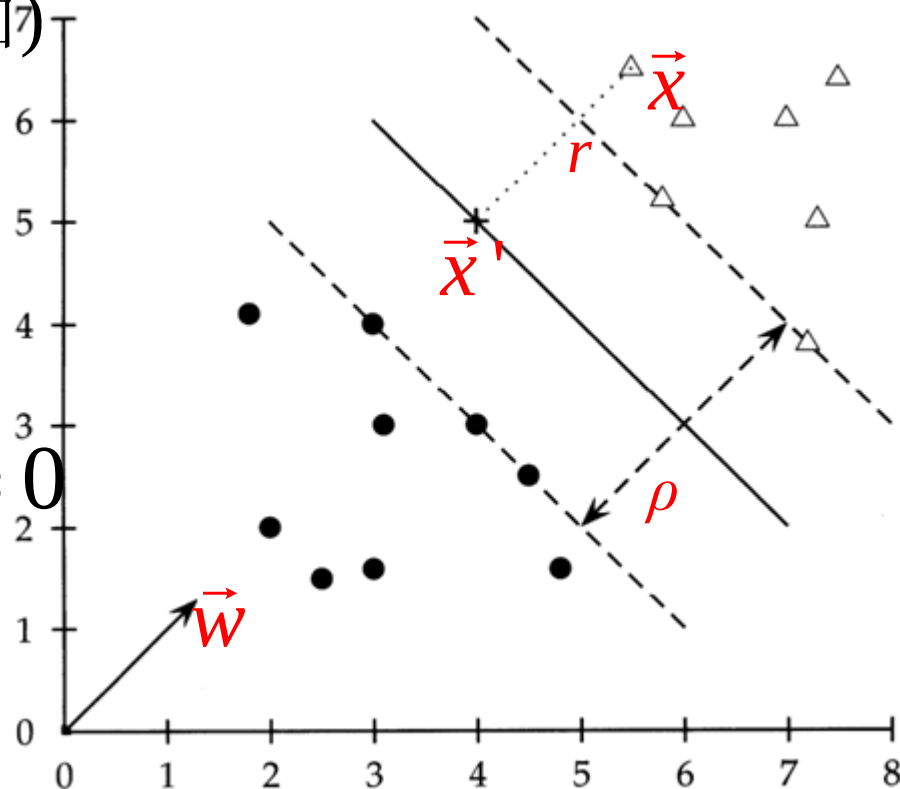
$$\vec{x}' = \vec{x} - y * r * \vec{w} / |\vec{w}|$$

- $\vec{x}'$ 在超平面上，所以满足

$$\vec{w}^T \vec{x}' + b = 0$$

$$\vec{w}^T (\vec{x} - y * r * \vec{w} / |\vec{w}|) + b = 0$$

- 所以 $r = y \frac{\vec{w}^T \vec{x} + b}{|\vec{w}|}$



线性支持向量机

- 线性可分的情况
- 假设所有数据离超平面的最小距离是1，那么对于一个训练集来说 $\{(\vec{x}_i, y_i)\}$ ，满足

$$\vec{w}^T \vec{x}_i + b \geq 1 \quad \text{if } y_i = 1$$

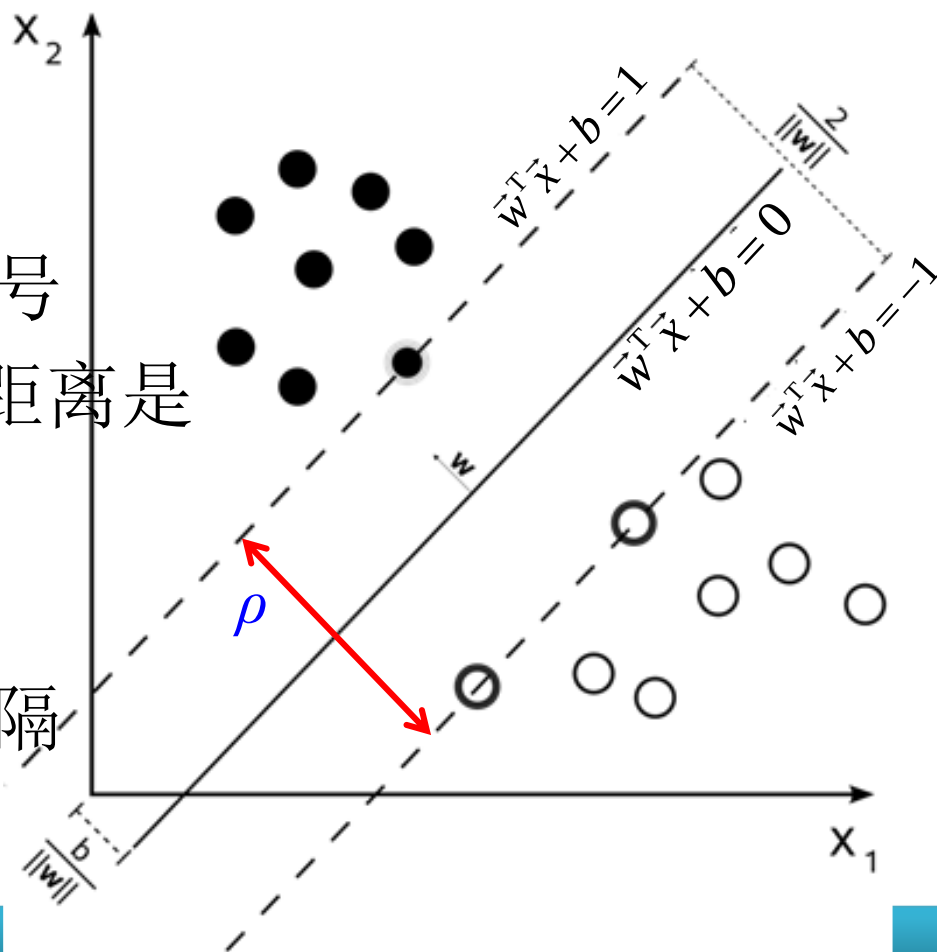
$$\vec{w}^T \vec{x}_i + b \leq -1 \quad \text{if } y_i = -1$$

- 对支持向量来说上式是等号
- 每个样本 $\vec{x}$ 点到超平面的距离是

$$r = y \frac{\vec{w}^T \vec{x} + b}{|\vec{w}|}$$

- 把支持向量代入，那么间隔

$$\rho = \frac{2}{|\vec{w}|}$$



- 超平面 $\vec{w}^T \vec{x}_i + b = 0$
- 约束条件：样本点到超平面的最小距离是1

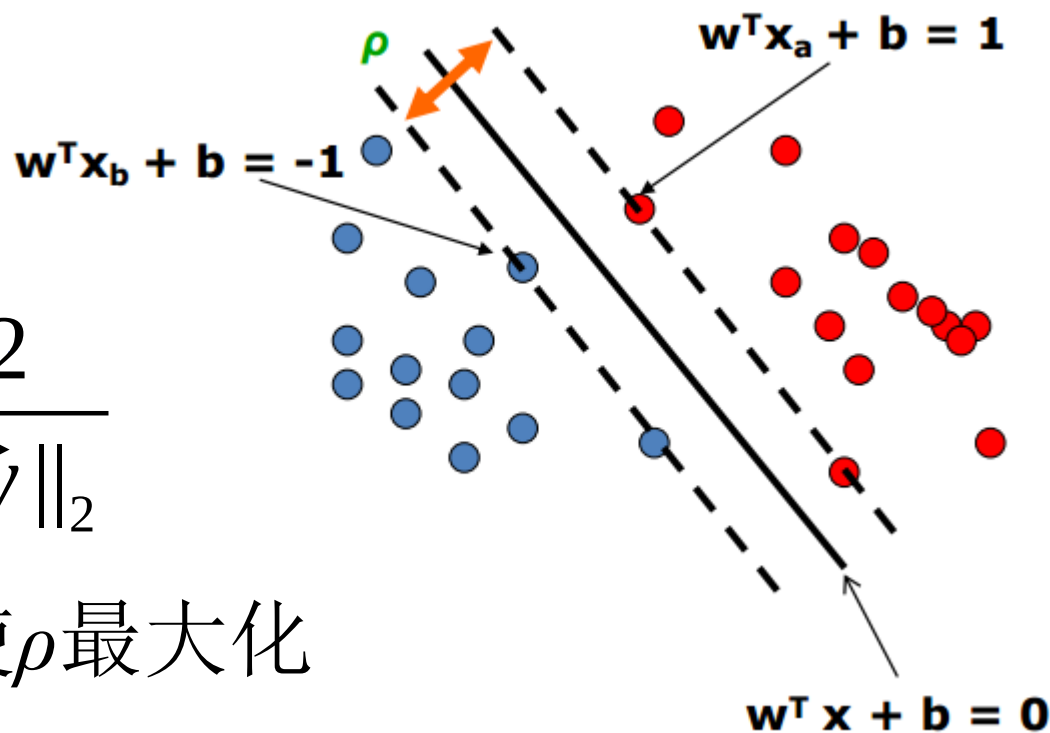
$$\min_{i=1,\dots,n} |\vec{w}^T \vec{x}_i + b| = 1$$

- 表明：

$$\vec{w}^T (\vec{x}_a - \vec{x}_b) = 2$$

$$\rho = \|\vec{x}_a - \vec{x}_b\|_2 = \frac{2}{\|\vec{w}\|_2}$$

- 支持向量机就是使 ρ 最大化
(即间隔最大化)



- 那么这个问题就变为一个凸优化问题：
 - 寻找 $\vec{w}$ 和 b ，使得

$$\rho = \frac{2}{\|\vec{w}\|} \text{ 最大化 (等价于 } \frac{\|\vec{w}\|}{2} \text{ 最小化)，并且所}$$
- 有 $\{(\vec{x}_i, y_i)\}$ ，有

$$\vec{w}^T \vec{x}_i + b \geq 1 \text{ if } y_i = 1; \vec{w}^T \vec{x}_i + b \leq -1 \text{ if } y_i = -1$$

(既 $y_i(\vec{w}^T \vec{x}_i + b) \geq 1$)
- 更好的描述
 - 寻找 $\vec{w}$ 和 b ，使得

$$\Phi(\vec{w}) = \frac{1}{2} \vec{w}^T \vec{w} \text{ 最小化，}$$

并且所有 $\{(\vec{x}_i, y_i)\}$ ，有 $y_i(\vec{w}^T \vec{x}_i + b) \geq 1$

- 上述问题实际上是在线性约束条件下的二次函数优化问题。该问题是数学中的基本问题之一，存在很多解决算法。本文通过拉格朗日对偶的方法求解。
- 求解结果
 - 对于训练集中的每个约束条件 $y_i(\vec{w}^T \vec{x}_i + b) \geq 1$ 都对应一个拉格朗日因子 α_i
 - 那么求得：

$$\vec{w} = \sum \alpha_i y_i \vec{x}_i$$

$$b = y_k - \vec{w}^T \vec{x}_k \text{ for any } x_k \text{ such that } \alpha_k \neq 0$$

- 每个非零 α_i 表示对应的 $\vec{x}_i$ 是一个支持向量
- 最终的分类器：

$$f(\vec{x}) = \text{sign}(\sum \alpha_i y_i \vec{x}_i^T \vec{x} + b)$$

小结：SVM要点

- 线性SVM分类器为

$$f(\vec{x}) = \text{sign}(\sum \alpha_i y_i \vec{x}_i^T \vec{x} + b)$$

- SVM基本过程

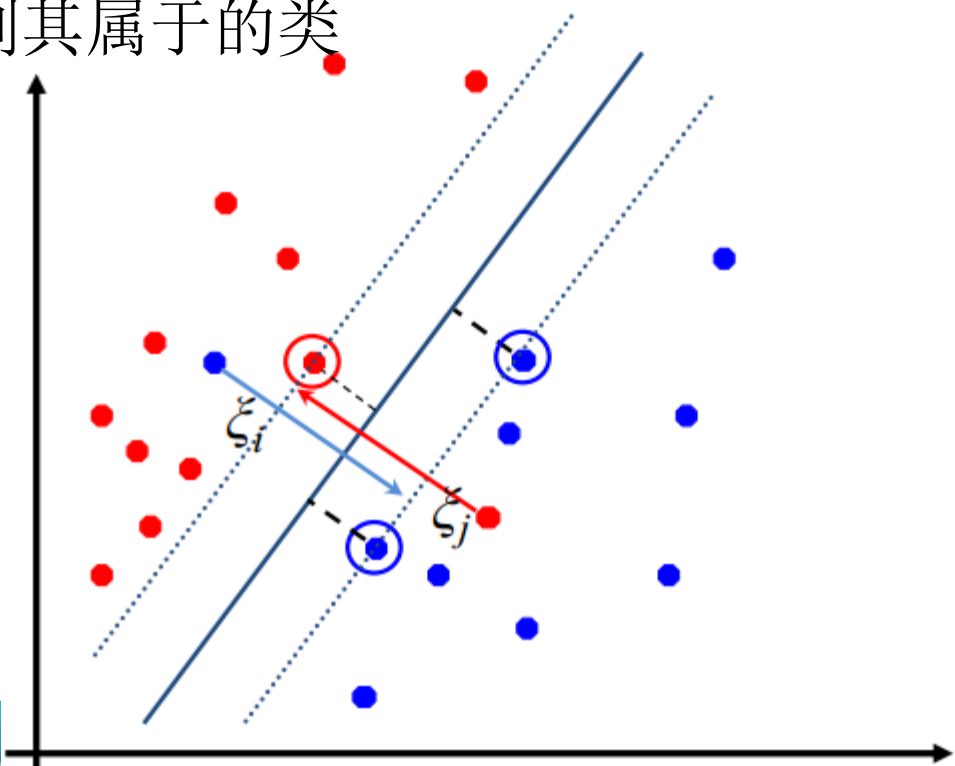
- 基于给定训练数据集，通过二次优化过程寻找最佳的分类超平面
- 对于待分类的新数据点，利用分类函数计算该点到超平面的距离
- 距离的正负(分类函数的符号)决定了该数据点类别的归属
- 如果该点在分类器的间隔之内，分类器可以在原来的两个类之外，返回“类别未知”

内容

- 二元线性SVM
- SVM用于非线性分类

软间隔(Soft margin)分类

- 如果训练集不是线性可分，引入松弛变量(slack variables) ξ_i 来允许决策间隔犯错误
 - 非零 ξ_i ，表示允许 $\vec{x}_i$ 在不满足间隔需求下的惩罚量或代价因子
- 允许错误发生
 - 以一定的代价把噪音点移到其属于的类
- 训练的目标依然是最小化错误，并求最大间隔超平面



- 旧形式

寻找 $\vec{w}$ 和 b , 使得:

$$\Phi(\vec{w}) = \frac{1}{2} \vec{w}^T \vec{w} \text{ 最小化,}$$

并且所有 $\{(\vec{x}_i, y_i)\}$, 有 $y_i(\vec{w}^T \vec{x}_i + b) \geq 1$

- 软间隔分类新形式

寻找 $\vec{w}$ 、 b 及 $\xi_i > 0$, 使得:

$$\Phi(\vec{w}) = \frac{1}{2} \vec{w}^T \vec{w} + C \sum \xi_i \text{ 最小化,}$$

并且所有 $\{(\vec{x}_i, y_i)\}$, 有 $y_i(\vec{w}^T \vec{x}_i + b) \geq 1 - \xi_i$

- 软间隔分类新形式

寻找 $\vec{w}$ 、 b 及 $\xi_i > 0$ ，使得：

$$\Phi(\vec{w}) = \frac{1}{2} \vec{w}^T \vec{w} + C \sum \xi_i \text{ 最小化,}$$

并且所有 $\{(\vec{x}_i, y_i)\}$ ，有 $y_i(\vec{w}^T \vec{x}_i + b) \geq 1 - \xi_i$

- 优化问题就转化为间隔的宽度和允许进入间隔内的数据点数目之间的均衡性问题。

- 对于点 $\vec{x}_i$ 来说，通过设置 $\xi_i > 0$ ，间隔的宽度可以小于1，但是在最小化过程中要付出惩罚代价 $C\xi_i$ 。 ξ_i 的和是训练错误的上界。

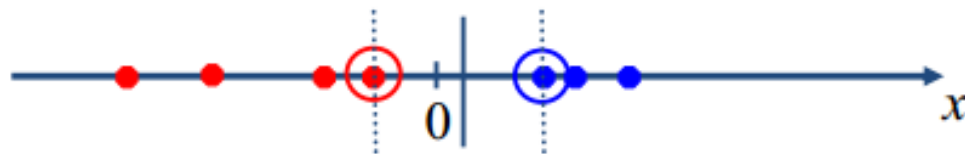
- 基于软间隔的SVM在训练错误和间隔大小之间权衡，并使目标函数最小化。

- C 是正则化(regularization)因子，控制过拟合问题：

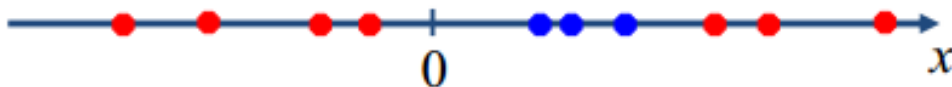
- 如果 C 变大，会对出现在间隔内的点的惩罚较大，即更尊重数据本身，当然这时候的代价就是减小几何间隔。
 - 当 C 很小时，则很容易通过松弛变量来考虑噪音点，此时可以得到能够对大部分数据点建模的更宽的间隔。

非线性SVM

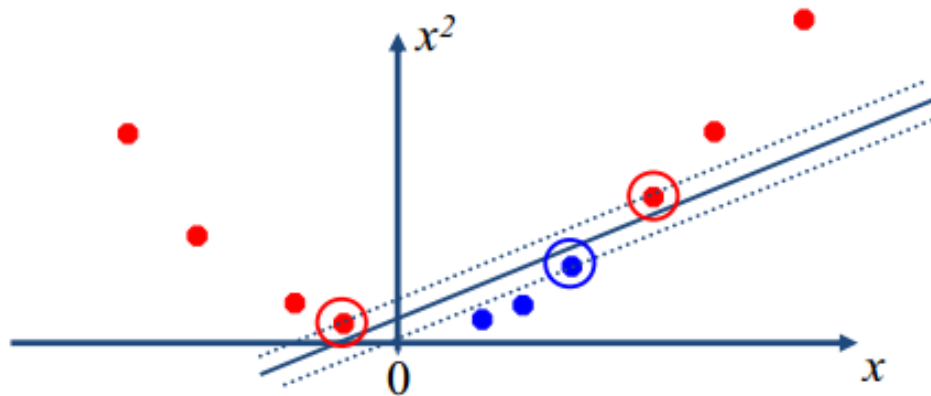
- 数据集是线性可分的(最多包括少数例外点或噪音点)



- 数据集不可能被线性分类器直接分开?

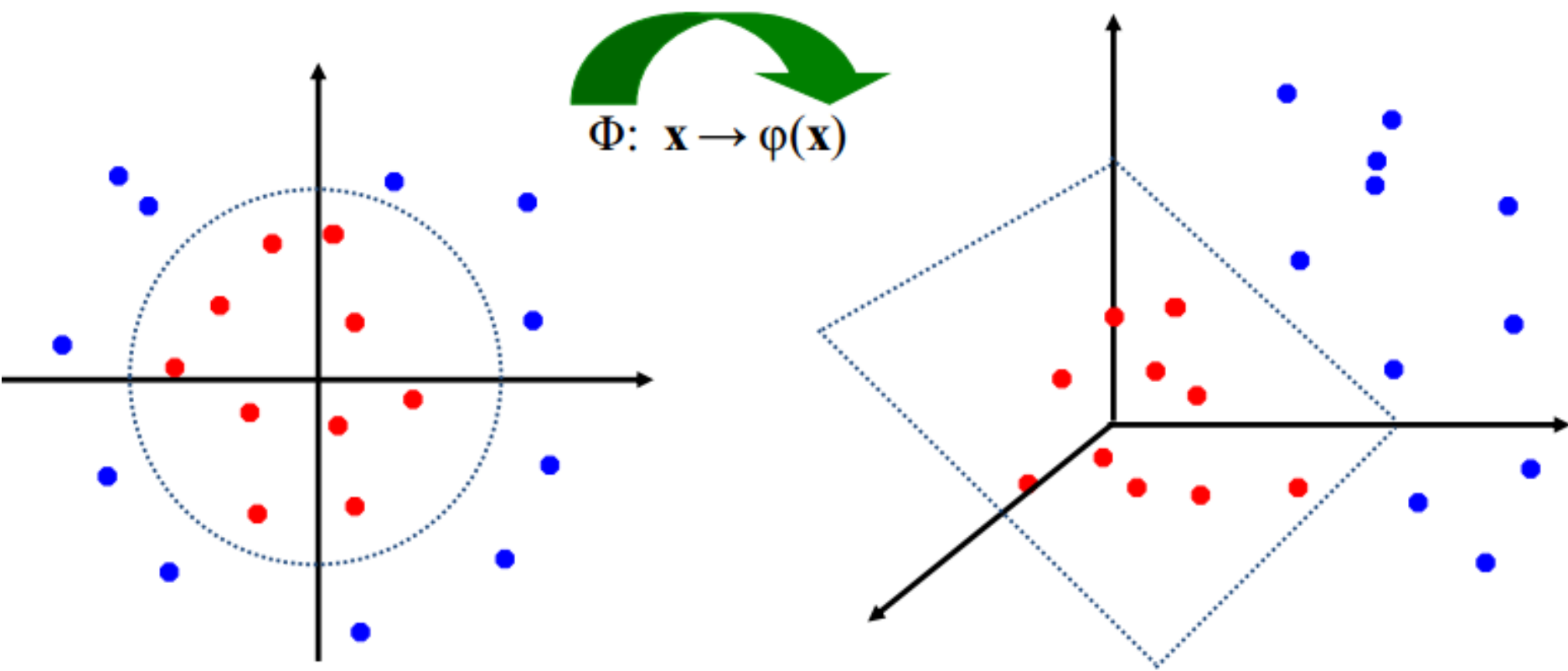


- 解决方法: 将数据映射到一个高维空间并在此空间上使用线性分类器将数据分开



非线性SVM：特征空间

- 一般思路：将原始的特征空间映射到某个更高维的线性可分的特征空间上去



核技巧(kernel trick)

- SVM 线性分类器依赖于数据点之间的内积操作。令

$$K(\vec{x}_i, \vec{x}_j) = \vec{x}_i^T \vec{x}_j$$

- 通过某个映射函数 Φ 可以将原始空间的点映射到新空间，即 $\Phi: \vec{x} \mapsto \Phi(\vec{x})$ 那么，新空间下的点积计算变为

$$K(\vec{x}_i, \vec{x}_j) = \Phi(\vec{x}_i)^T \Phi(\vec{x}_j)$$

- 核函数 K 实际上是一个对应于新特征空间上的内积函数
- 如果能够证明该点积(也就是一个实数)能够通过原始数据点简单高效地计算出来，那么就不必真的要将 $\vec{x} \mapsto \Phi(\vec{x})$ 。相反，可以直接通过 $K(\vec{x}_i, \vec{x}_j) = \Phi(\vec{x}_i)^T \Phi(\vec{x}_j)$ 来计算。
- 那么线性分类器 $\rightarrow$ 非线性分类器

$$f(\vec{x}) = \text{sign}(\sum \alpha_i y_i \vec{x}_i^T \vec{x} + b)$$

$$\Rightarrow f(\vec{x}) = \text{sign}(\sum \alpha_i y_i K(\vec{x}_i^T \vec{x}) + b)$$

二维空间上一个二次核函数的例子

- 对于二维的向量 $\vec{u} = (u_1, u_2)$ $\vec{v} = (v_1, v_2)$
- 考虑 $K(\vec{u}, \vec{v}) = (1 + \vec{u}^T \vec{v})^2$
- 下面证明这是一个核函数，也就是说，对于某个函数 Φ ，有 $K(\vec{u}, \vec{v}) = \Phi(\vec{u})^T \Phi(\vec{v})$
- 考虑： $\Phi(\vec{u}) = (1 \ u_1^2 \ \sqrt{2}u_1u_2 \ u_2^2 \ \sqrt{2}u_1 \ \sqrt{2}u_2)$
- 有

$$\begin{aligned} K(\vec{u}, \vec{v}) &= (1 + \vec{u}^T \vec{v})^2 \\ &= 1 + u_1^2 v_1^2 + 2u_1 v_1 u_2 v_2 + u_2^2 v_2^2 + 2u_1 v_1 + 2u_2 v_2 \\ &= (1 \ u_1^2 \ \sqrt{2}u_1u_2 \ u_2^2 \ \sqrt{2}u_1 \ \sqrt{2}u_2)^T (1 \ v_1^2 \ \sqrt{2}v_1v_2 \ v_2^2 \ \sqrt{2}v_1 \ \sqrt{2}v_2) \\ &= \phi(\vec{u})^T \phi(\vec{v}) \end{aligned}$$

小结：非线性SVM与Kernels

- 软间隔分类

寻找 $\vec{w}$ 、 b 及 $\xi_i > 0$ ，使得：

$$\Phi(\vec{w}) = \frac{1}{2} \vec{w}^T \vec{w} + C \sum \xi_i \text{ 最小化,}$$

并且所有 $\{(\vec{x}_i, y_i)\}$ ，有 $y_i(\vec{w}^T \vec{x}_i + b) \geq 1 - \xi_i$

- 非线性SVM

- 通过空间映射将原始空间映射到新空间，为避免显式的映射函数，引入核函数(定义在原始空间下但是结果是新空间下的内积函数)

- 常用核函数

- Linear

- Polynomial (多项式核) $K(\vec{x}, \vec{z}) = (1 + \vec{x} \cdot \vec{z})^d$

- Radial basis function (径向基核) $K(\vec{x}, \vec{z}) = e^{-(\vec{x} - \vec{z}) / (2\sigma^2)}$

本讲要点回顾

- 基于向量空间的分类
 - Rocchio
 - kNN
- 线性分类器-hyperplane
 - Rocchio
 - Naive Bayes
- **SVM**（支持向量机）
 - 线性SVM
 - 非线性SVM