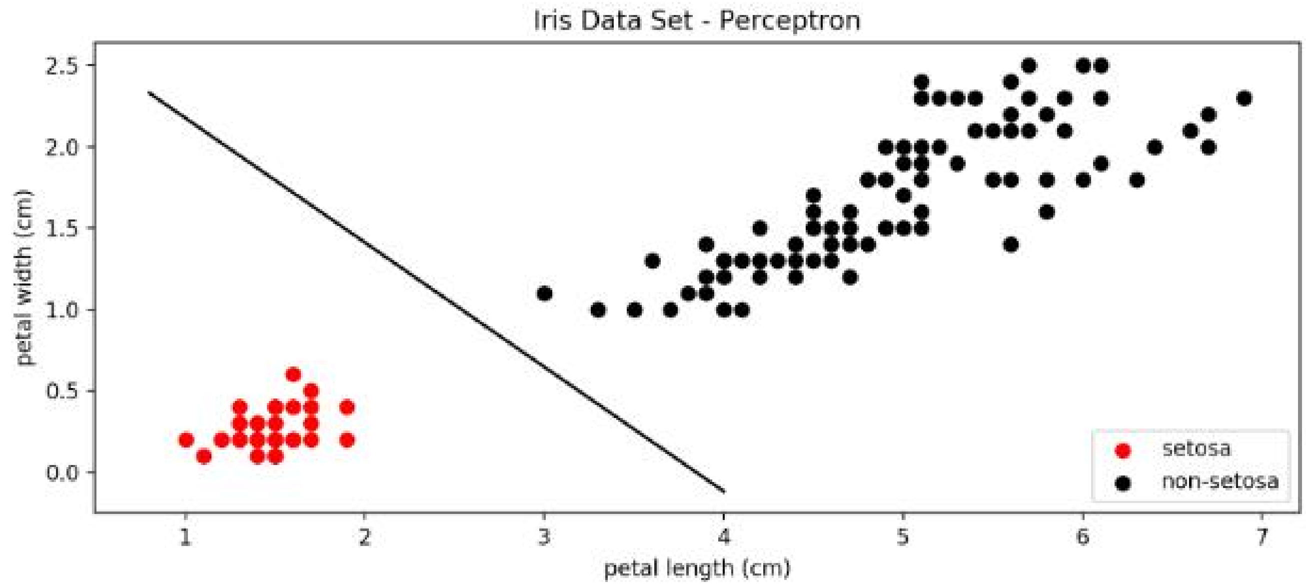
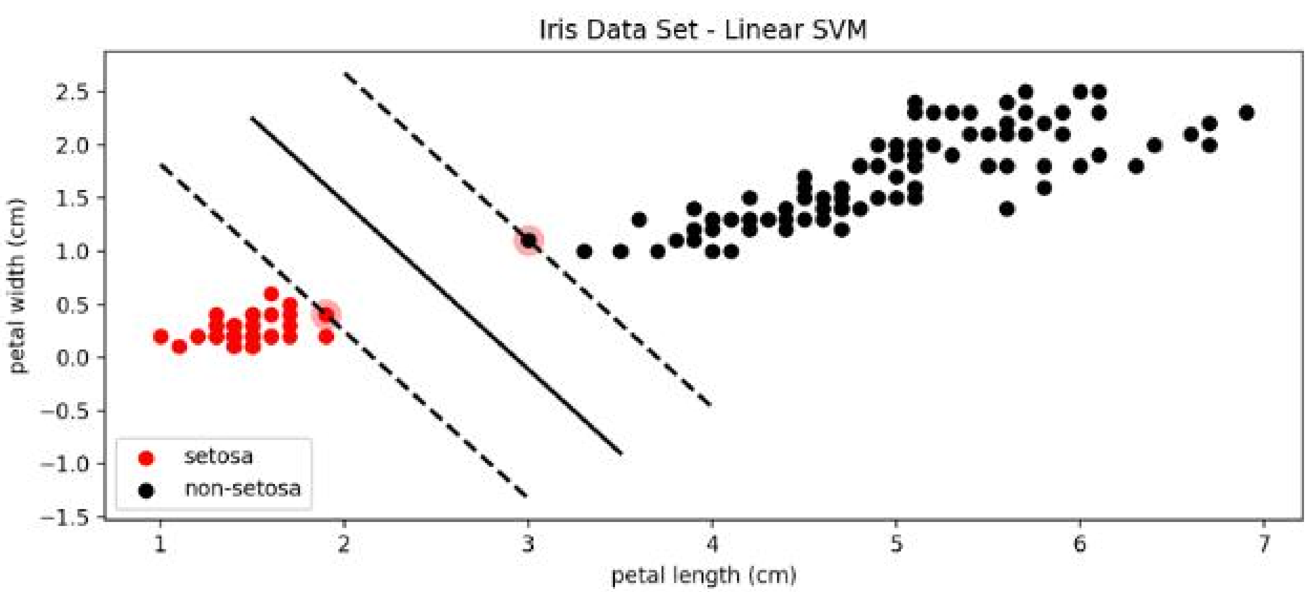
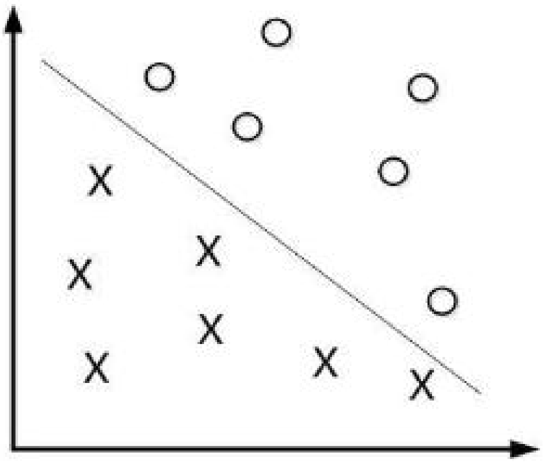
SVM学习笔记

1.1 分类函数
1.2 最大化分类间隔
1.3 原始问题转化为对偶问题求解
1.4 特征空间隐式映射：核函数
1.5 软间隔最大化目标函数的优化
1.6 支持向量回归SVR

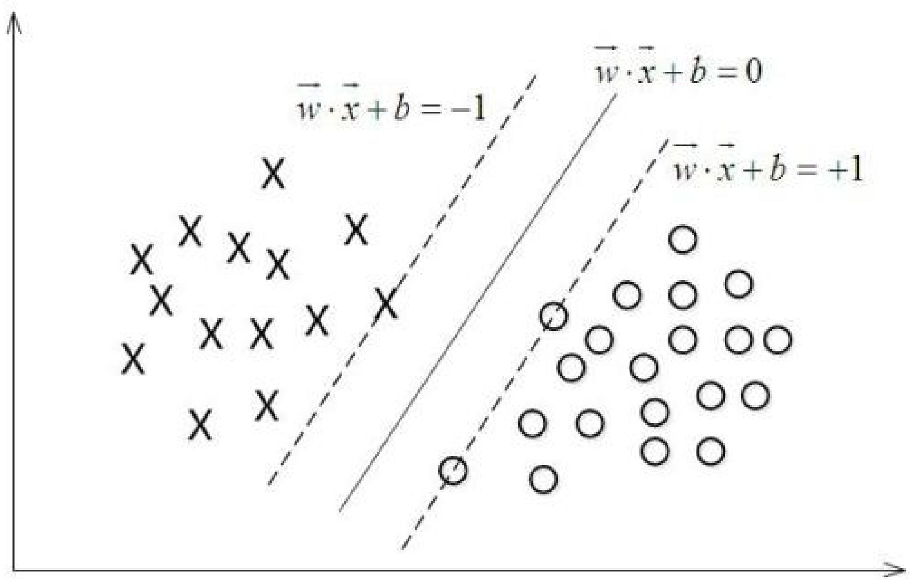

Iris Data Set - Perceptron
setosa
non-setosa
petal width (cm)
petal length (cm)
0.0
0.5
1.0
1.5
2.0
2.5
1
2
3
4
5
6
7


Iris Data Set - Linear SVM
setosa
non-setosa
petal width (cm)
petal length (cm)
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
2.5
1
2
3
4
5
6
7

1.1 分类函数

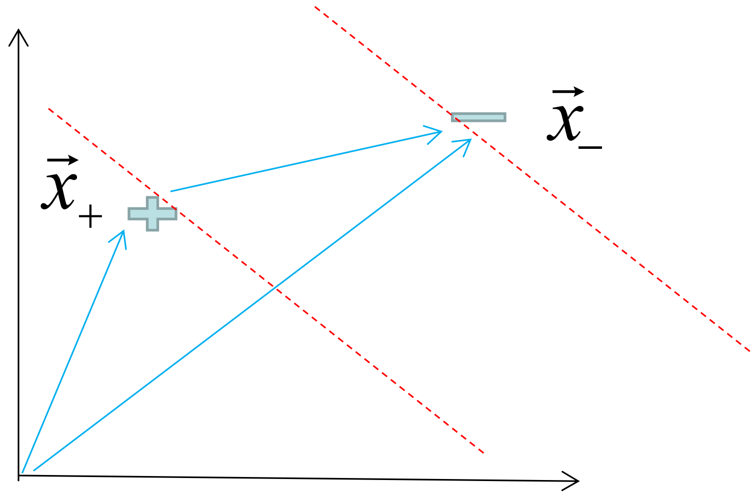


分类函数$f(x) = \mathrm{w}^{\mathrm{T}}x + b$



对函数值-1，+1的理解：可为任意值，为简化计算时使用

1.2最大化分类间隔



$$“路宽” = (\vec{x}_- - \vec{x}_+)\cdot\frac{\vec{w}}{|w|} = \frac{1}{|w|}(\vec{w}\cdot\vec{x}_- - \vec{w}\cdot\vec{x}_+) = \frac{2}{|w|}$$

$$\max\frac{2}{\|w\|} \to \max\frac{1}{\|w\|} \to \min\|w\| \to \min\frac{1}{2}\|w\|^2$$

1.3原始问题转化为对偶问题求解

$$\min \| w \|^2 \quad s.t. \quad y_i(\mathrm{w}^{\mathrm{T}} x_i + b) \geq 1,\ \mathrm{i} = 1,2...\mathrm{n}$$

利用拉格朗日乘子法

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

$\alpha = (\alpha_1, \alpha_2 ... \alpha_n)^T$为拉格朗日乘子向量，令

$$\theta(w) = \max_{\alpha_i \geq 0} \mathcal{L}(w,b,\alpha)$$

目标函数：

$$\min_{w,b} \theta(w) = \min_{w,b} \max_{\alpha_i \geq 0} \mathcal{L}(w, b, \alpha) = p^*$$

$$\max_{\alpha_i \geq 0} \min_{w,b} \mathcal{L}(w, b, \alpha) = d^*$$

交换之后，变为原问题的对偶问题。经验证，满足KKT条件，两者相等 $d^* = p^*$

对偶问题求解：

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

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

带入原函数得到： $L(w,b,a) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^T x_j$

对 α 的极大，即是关于对偶问题的最优化问题。

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

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

求出α_i得到

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

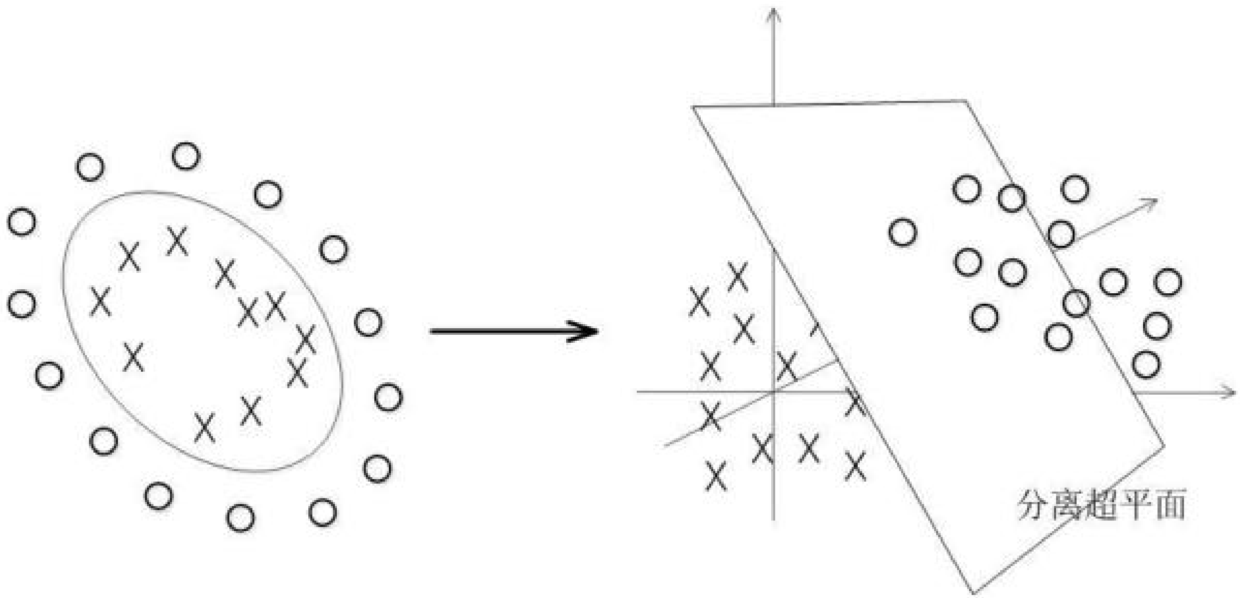
$$b^* = y_j - \sum_{i=1}^{N} y_i \alpha_i^* (x_i \cdot x_j)$$

得到分类函数（线性学习分类器）：

$$f(x) = (\sum_{i=1}^{n} \alpha_i y_i x_i)^T x + b = \sum_{i=1}^{n} \alpha_i y_i < x_i, x > + b$$

1.4特征空间隐式映射：核函数

定义 3 (核:Kernel) 核是一个函数 K，对所有 $x, z \in \mathcal{X}$，满足 $K(x, z) = \langle \phi(x), \phi(z) \rangle$，这里 ϕ 是从 $\mathcal{X}$ 到内积特征空间 $\mathcal{F}$ 的映射。



核函数的价值在于它虽然也是讲特征进行从低维到高维的转换，但核函数事先在低维上进行计算，而将实质上的分类效果表现在了高维上，避免了直接在高维空间中的复杂计算。

核函数将原来的分类函数 $$f(x)=\sum_{i=1}^{n}\alpha_i y_i\langle x_i, x\rangle+b$$

映射成：

$$f(x)=\sum_{i=1}^{n}\alpha_i y_i\langle\phi(x_i),\phi(x)\rangle+b$$

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

α 可以通过求解如下 dual 问题得到，

$$\begin{aligned}\max_{\alpha}\quad & \sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j\langle\phi(x_i),\phi(x_j)\rangle\\ s.t.\quad & \alpha_i\geq 0, i=1,\ldots,n\\ & \sum_{i=1}^{n}\alpha_i y_i=0\end{aligned}$$

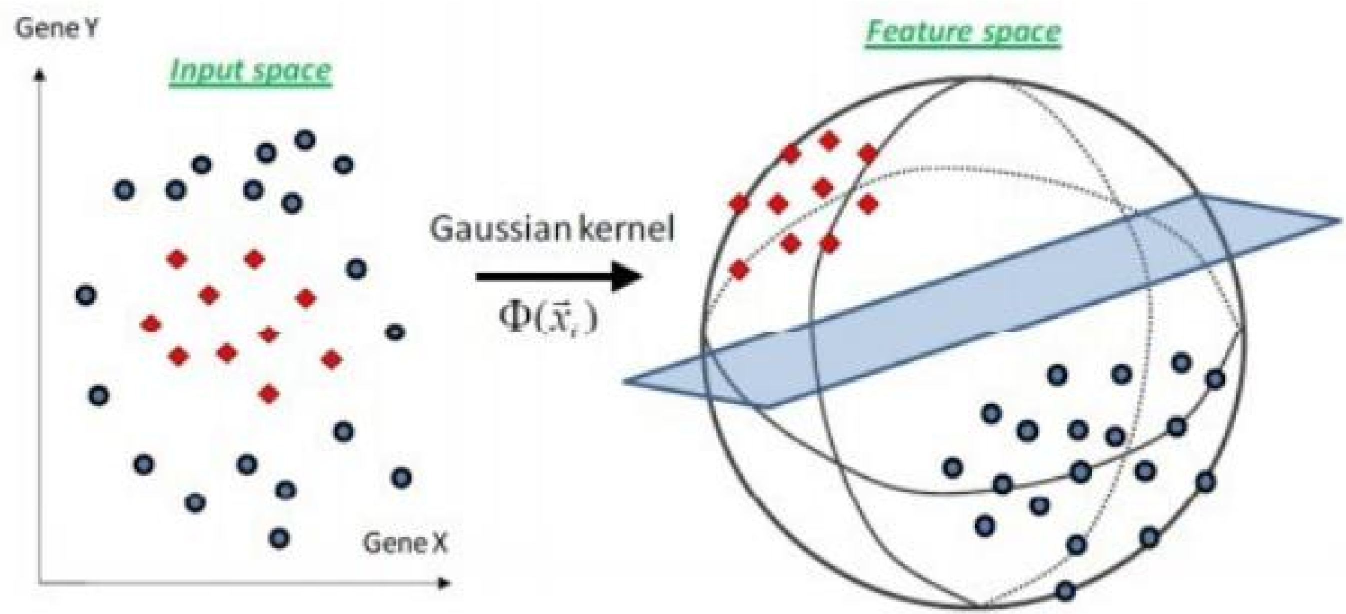
$$\begin{aligned}\max_{\alpha}\quad & \sum_{i=1}^{n}\alpha_i-\frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j K(x_i,x_j)\\ s.t.\quad & \alpha_i\geq 0, i=1,\ldots,n\\ & \sum_{i=1}^{n}\alpha_i y_i=0\end{aligned}$$

计算两个向量在隐式映射过后的空间中的内积的函数叫做核函数(Kernel Function)。

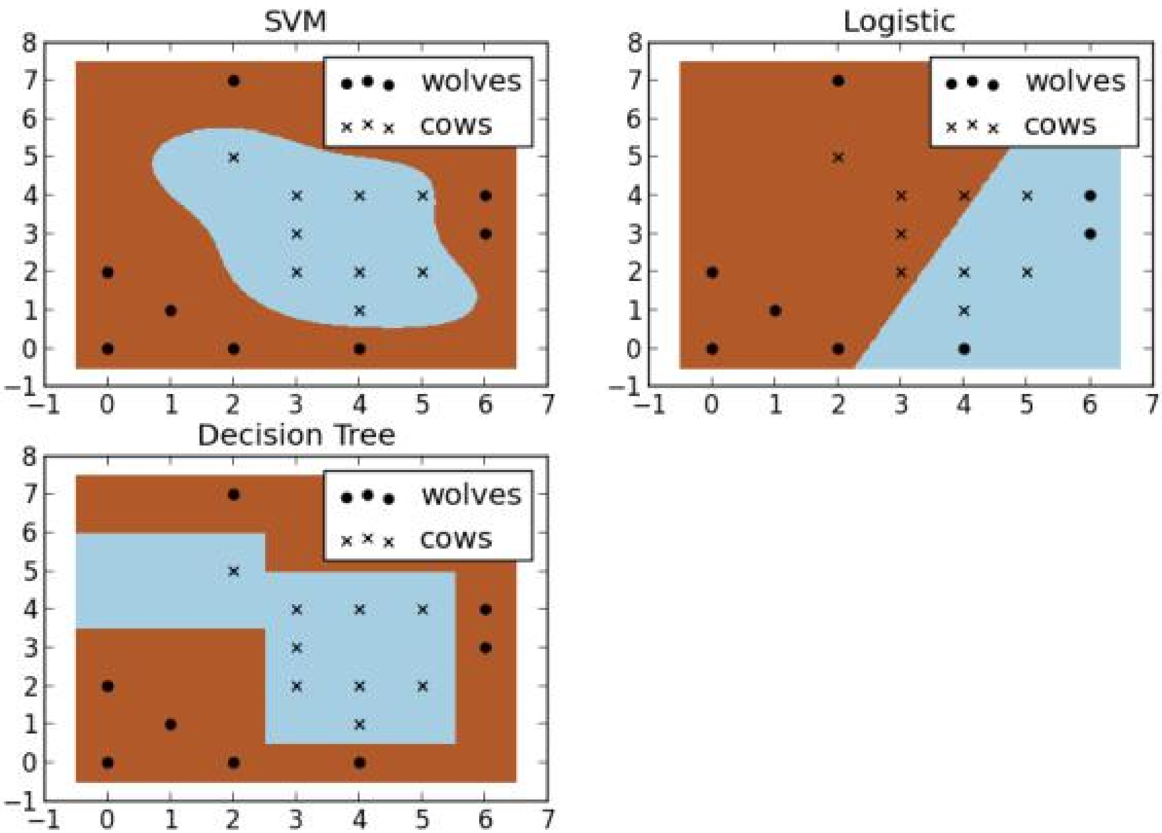
多项式核 $K(x_1, x_2) = (\langle x_1, x_2 \rangle + R)^d$

高斯核 $K(x_1, x_2) = exp(-\|x_1 - x_2\|^2 / 2\sigma^2)$

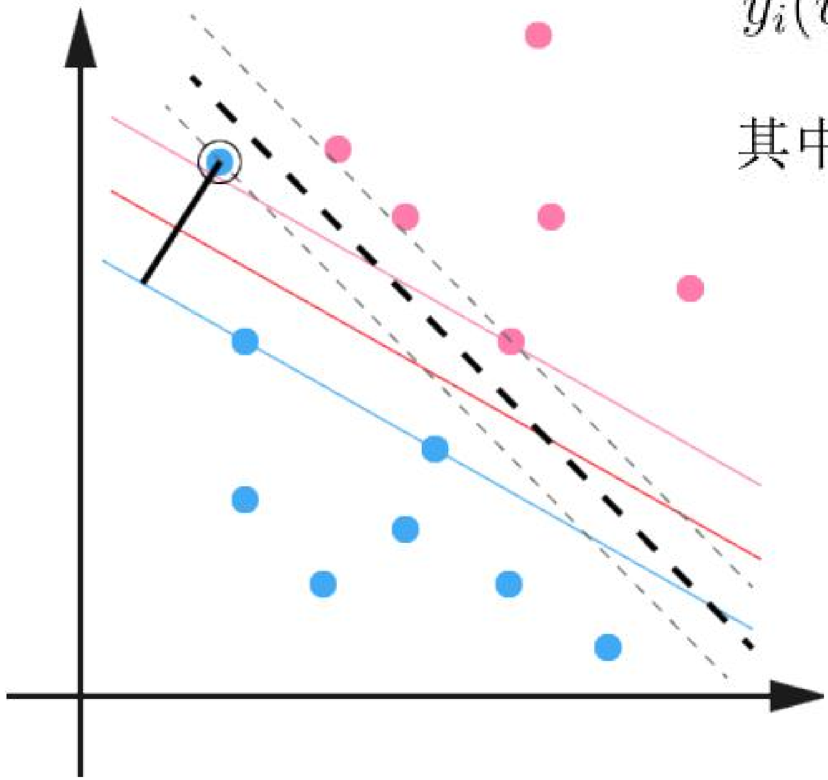
σ 选得很大的话，高次特征上的权重实际上衰减得非常快，所以实际上（数值上近似一下）相当于一个低维的子空间；如果 σ 选得很小，则可以将任意的数据映射为线性分—当然，随之而来的可能是非常严重的过拟合问题。



比较几种分类器，分类结果如下



1.5软间隔最大化目标函数的优化



$$y_i(w^T x_i + b) \geq 1 - \xi_i, \quad i = 1, ..., n$$

其中 ξ_i 称为松弛变量 (slack variable)

$$\min \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i$$

$$s.t. \quad y_i(w^T x_i + b) \geq 1 - \xi_i, i = 1, ..., n$$

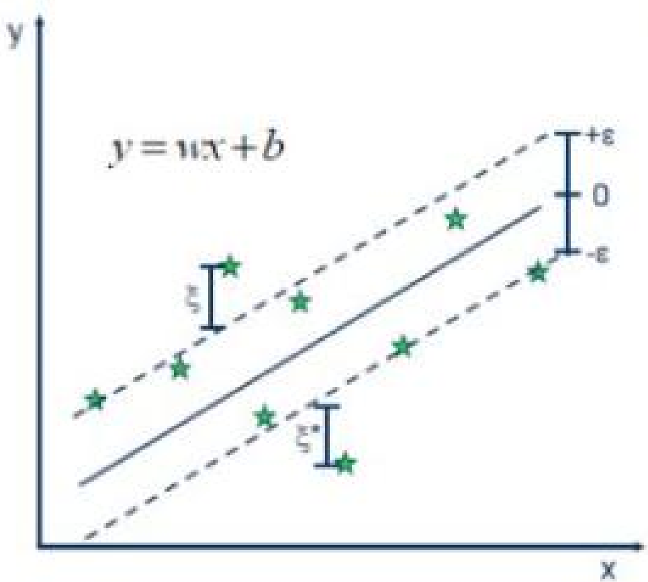
$$\xi_i \geq 0, i = 1, ..., n$$

模型复杂度过高，模型**过拟合**。虽然训练出来的模型能够在训练集上表现很好，但其**泛化能力会很差**。

1.6支持向量回归SVR

SVR：输出 wx+b，即某个样本点到分类面的距离，是连续值，所以是回归模型。

SVM：把这个距离用 sign(·) 函数作用，距离为正(在超平面一侧)的样本点是一类，为负的是另一类，所以是分类模型。



• Minimize:

$$\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*)$$

• Constraints:

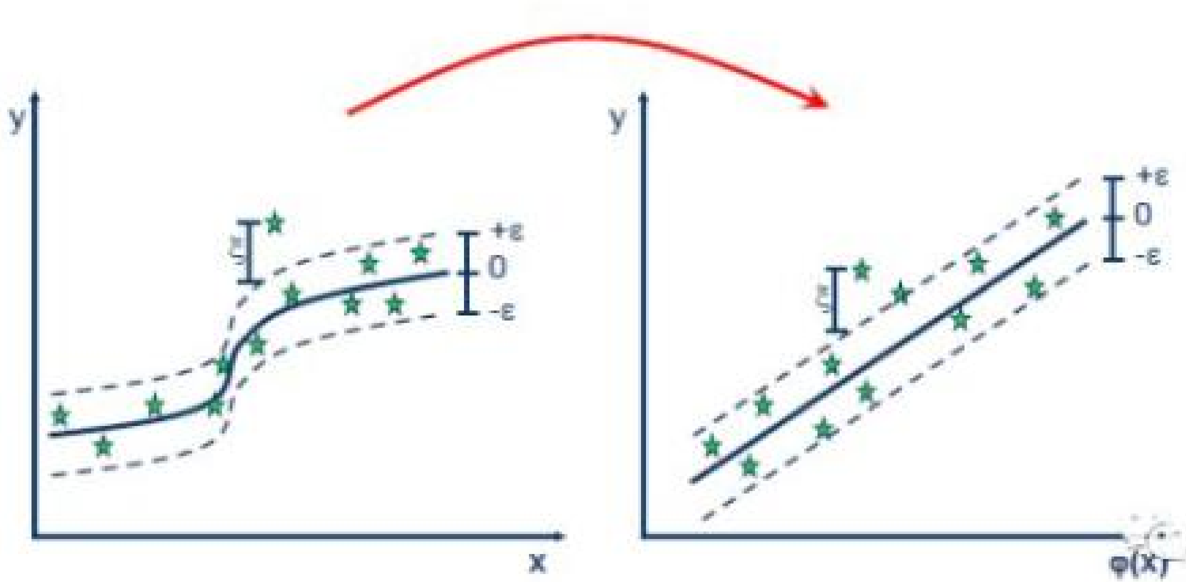
$$y_i - wx_i - b \le \varepsilon + \xi_i$$

$$wx_i + b - y_i \le \varepsilon + \xi_i^*$$

$$\xi_i, \xi_i^* \ge 0$$

最简单的线性回归模型是要找出一条曲线使得残差最小。同样的，SVR也是要找出一个超平面，使得所有数据到这个超平面的距离最小。

在SVR中，定义一个ε，定义虚线内区域的数据点的残差为0，而虚线区域外的数据点（支持向量）到虚线的边界的距离为残差（ζ）。与线性模型类似，使残差（ζ）最小。所以大致上来说，SVR就是要找出一个最佳的条状区域（2ε宽度），再对区域外的点进行回归。



对于非线性的模型，与SVM一样使用核函数（kernel function）映射到特征空间，然后再进行回归。