机器学习与数据挖掘

Machine Learning & Data Mining

权小军 教授

中山大学数据科学与计算机学院

quanxj3@mail.sysu.edu.cn

Preface

Preface

2006年12月，国际会议IEEE International Conference on Data Mining（ICDM）评选出了数据挖掘领域的十大经典算法

Preface

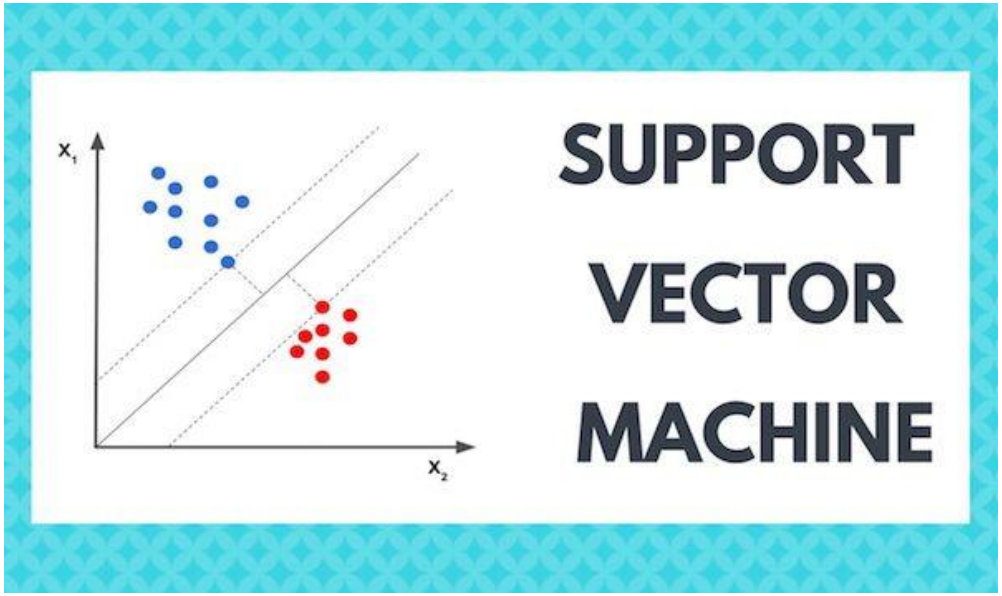
数据挖掘Top 10算法

1. C4.5
2. K-Means
3. SVM
4. Apriori
5. EM
6. PageRank
7. AdaBoost
8. kNN
9. Naive Bayes
10. CART

Preface

数据挖掘Top 10算法

1. C4.5
2. K-Means
3. SVM
4. Apriori
5. EM
6. PageRank
7. AdaBoost
8. kNN
9. Naive Bayes
10. CART



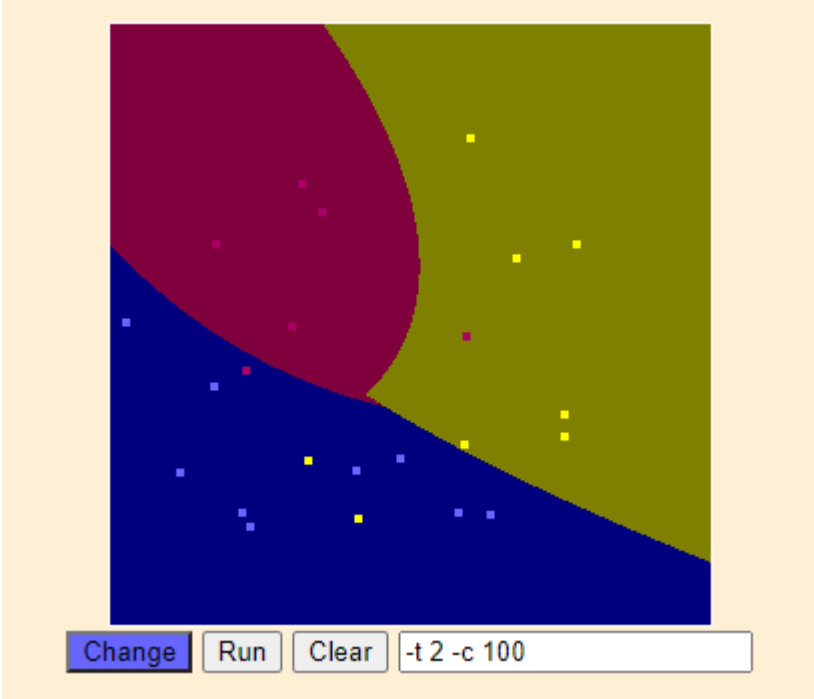
Preface

在机器学习领域, 有两大方法既理论优美, 又在实践中取得很好效果. 其中一类便是支持向量机及其衍生的核方法, 而另一类是AdaBoost及其衍生的Boosting方法. 例如, 在Kaggle竞赛中十分流行的xgboost即属于Boosting方法.

机器学习领域能有今天的地位, 这两大方法功不可没. 理解这两大方法, 对理解整个机器学习领域的发展脉络和思想体系有很大帮助. 现实世界的任务纷繁复杂, 如果只会用现有深度学习框架, 而不能做到"知其然, 又知其所以然", 那么对复杂任务很难有还手之力.

-知乎

Preface



https://www.csie.ntu.edu.tw/~cjlin/libsvm/

Lecture 9:
Support Vector Machines

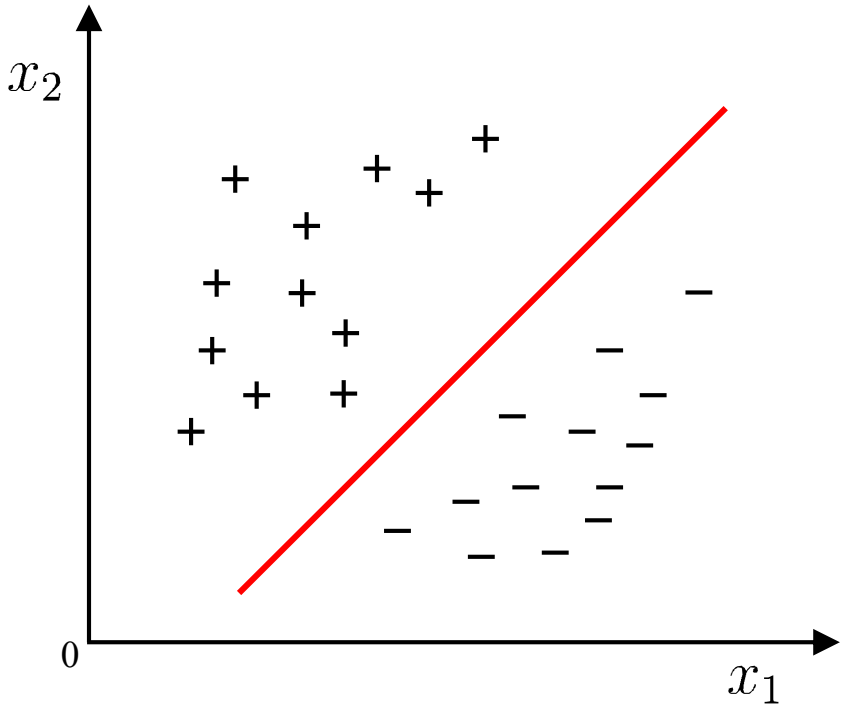
支持向量机(support vector machines)

- 支持向量机（support vector machines, SVM）是一种二分类模型，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

1. Margin and Support Vector

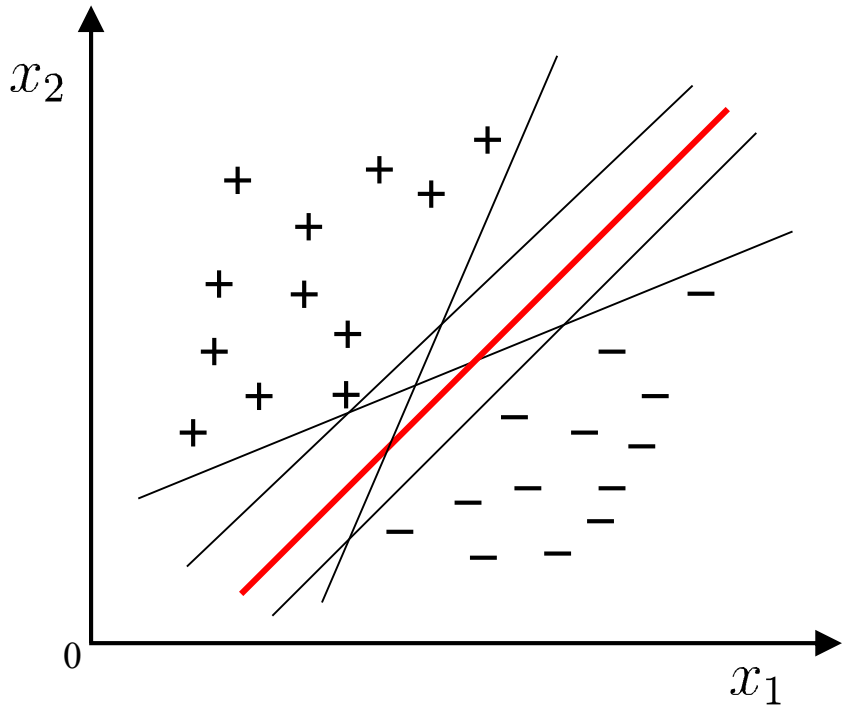
引子

线性模型：在样本空间中寻找一个超平面，将不同类别的样本分开.



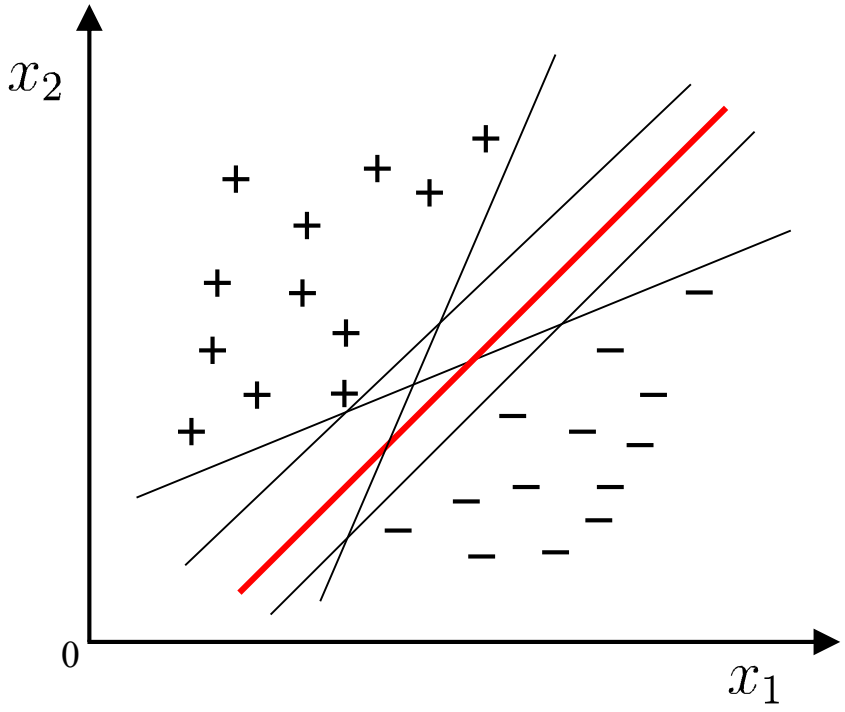
引子

-Q:将训练样本分开的超平面可能有很多, 哪一个好呢?



引子

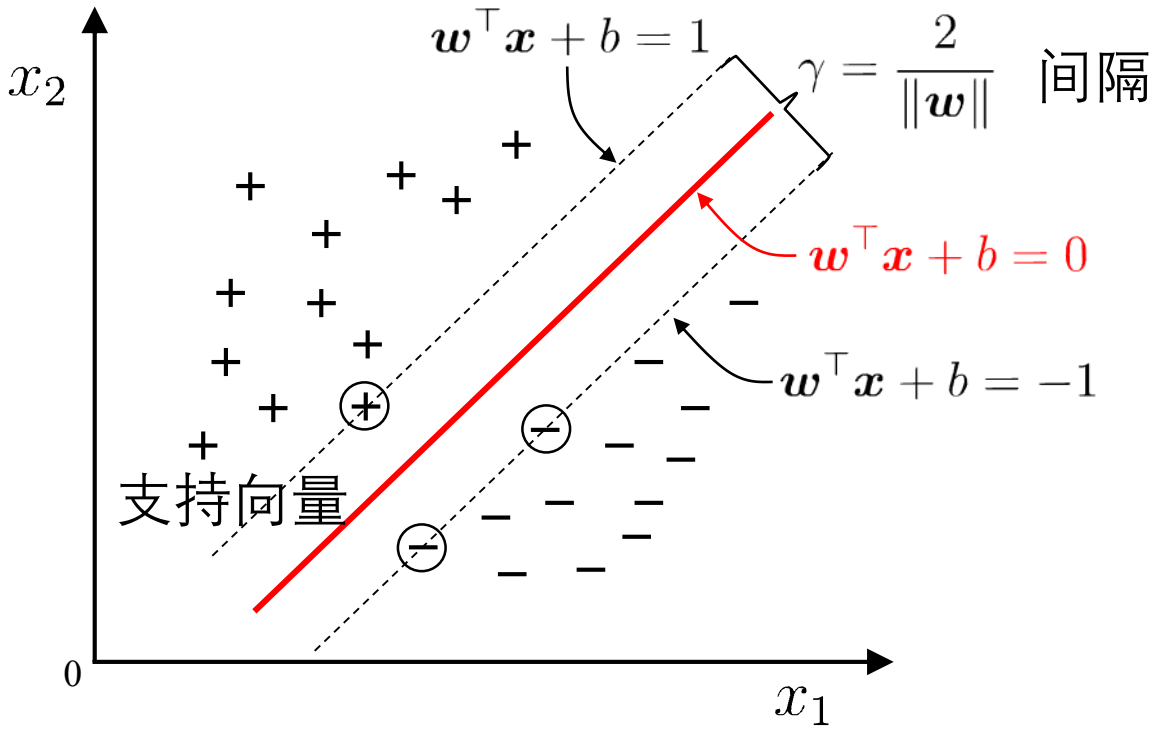
-Q:将训练样本分开的超平面可能有很多, 哪一个好呢?



-A:应选择"正中间", 容忍性好, 鲁棒性高, 泛化能力最强.

间隔与支持向量

超平面方程: $\boldsymbol{w}^\top \boldsymbol{x} + b = 0$



支持向量机基本型

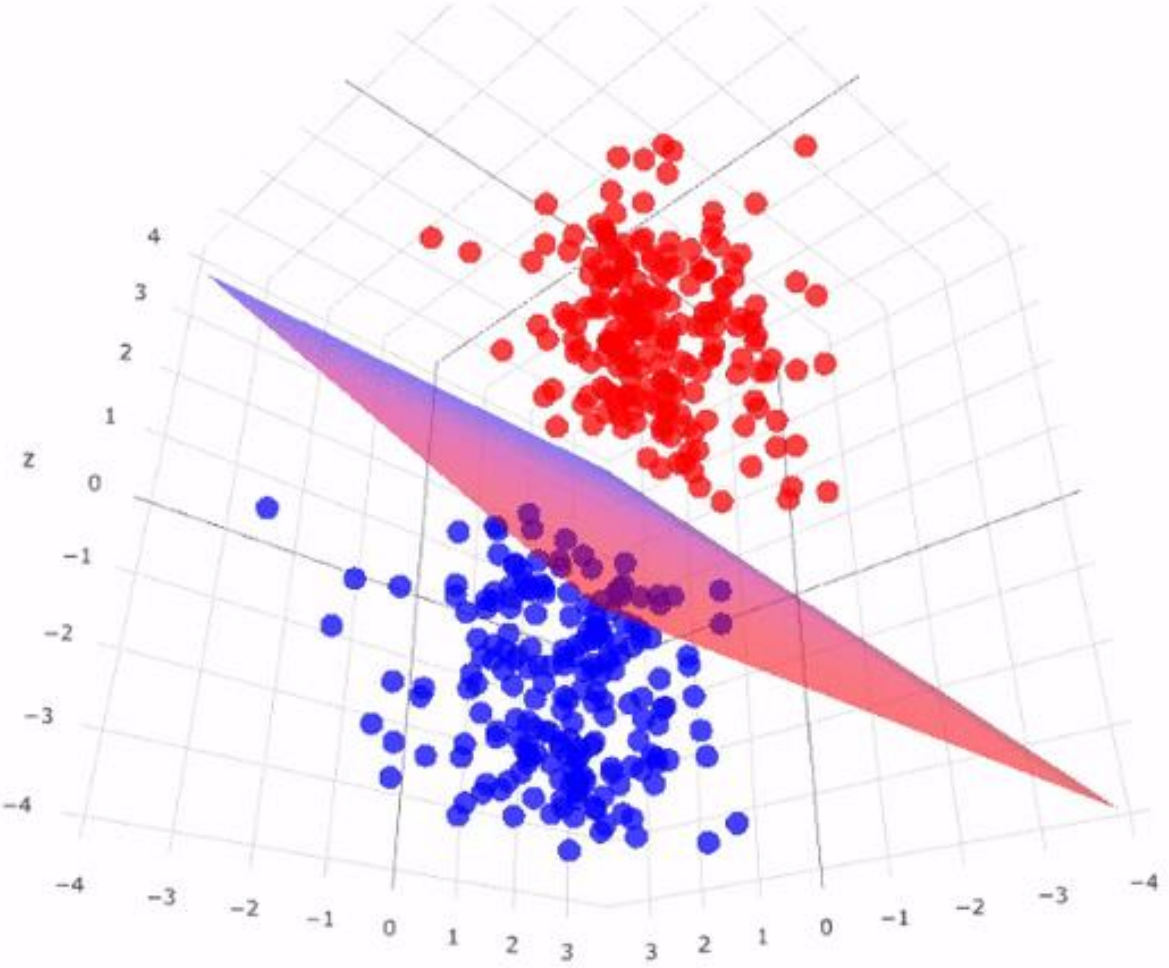
- 最大间隔: 寻找参数 $\boldsymbol{w}$ 和 b , 使得 γ 最大.

$$\begin{aligned} \arg\max_{\boldsymbol{w},b} \quad & \frac{2}{\|\boldsymbol{w}\|} \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1, \quad i = 1, 2, \ldots, m. \end{aligned}$$

$$\begin{aligned} \arg\min_{\boldsymbol{w},b} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1, \quad i = 1, 2, \ldots, m. \end{aligned}$$

支持向量机的基本模型

支持向量机基本型



2. 对偶问题

对偶问题

对偶问题是利用拉格朗日对偶性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。优点是：

- 对偶问题往往更易于求解
- 自然引入核函数，推广到非线性分类问题的求解

对偶问题

拉格朗日乘子法

- 拉格朗日乘子法是一种寻找多元函数在其变量受到一个或多个条件的约束时的极值的方法。这种方法可以将一个有n个变量与k个约束条件的最优化问题转换为一个解有n + k个变量的方程组的解的问题。
- 比如，要求 $f(x,y)$, 在 $g(x,y)=c$ 时的最大值时，我们可以引入新变量拉格朗日乘子 λ，这时我们只需要求下列拉格朗日函数的极值：

$$\mathcal{L}(x,y,\lambda)=f(x,y)+\lambda\cdot\Big(g(x,y)-c\Big)$$

- 一般地，对含n个变量和k个约束的情况，有：

$$\mathcal{L}\left(x_1,\ldots,x_n,\lambda_1,\ldots,\lambda_k\right)=f\left(x_1,\ldots,x_n\right)-\sum_{i=1}^{k}\lambda_i g_i\left(x_1,\ldots,x_n\right)$$

支持向量机

- 优化目标

$$\begin{aligned} \arg\min_{\boldsymbol{w},b} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1, \quad i = 1, 2, \ldots, m. \end{aligned}$$

凸二次优化问题

二次优化(Quadratic Programming)

- The objective of quadratic programming is to find an n-dimensional vector x, that will

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\mathbf{x}^{\mathrm{T}} Q\mathbf{x} + \mathbf{c}^{\mathrm{T}}\mathbf{x} \\ \text{subject to} \quad & A\mathbf{x} \preceq \mathbf{b}, \end{aligned}$$

对偶问题

KKT条件：$\begin{cases} \alpha_i \geq 0, \\ y_i f(\boldsymbol{x}_i) \geq 1, \\ \alpha_i (y_i f(\boldsymbol{x}_i) - 1) = 0. \end{cases}$

- 拉格朗日乘子法
  - 第一步：引入拉格朗日乘子$\alpha_i \geq 0$得到拉格朗日函数

$$L(\boldsymbol{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{m} \alpha_i \left(y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) - 1\right)$$

  - 第二步：令 $L(\boldsymbol{w}, b, \boldsymbol{\alpha})$对 $\boldsymbol{w}$ 和 b的偏导为零可得

$$\boldsymbol{w} = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i, \quad \sum_{i=1}^{m} \alpha_i y_i = 0.$$

  - 第三步：回代

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j - \sum_{i=1}^{m} \alpha_i$$

$$\text{s.t.} \ \sum_{i=1}^{m} \alpha_i y_i = 0, \ \alpha_i \geq 0, \ i = 1, 2, \ldots, m.$$

解的稀疏性

- 最终模型: $f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b = \sum_{i=1}^{m} \alpha_i y_i \boldsymbol{x}_i^\top \boldsymbol{x} + b$

支持向量机解的稀疏性: 训练完成后, 大部分的训练样本都不需保留, 最终模型仅与支持向量有关.

求解方法 - SMO

- 基本思路：不断执行如下两个步骤直至收敛.
  - 第一步：选取一对需更新的变量 α_i 和 α_j.
  - 第二步：固定 α_i 和 α_j 以外的参数, 求解对偶问题更新 α_i 和 α_j

- 仅考虑 α_i 和 α_j 时, 对偶问题的约束变为

$$\alpha_i y_i + \alpha_j y_j = -\sum_{k \neq i,j} \alpha_k y_k, \quad \alpha_i \geq 0, \quad \alpha_j \geq 0.$$

用一个变量表示另一个变量, 回代入对偶问题可得一个单变量的二次规划, 该问题具有闭式解.

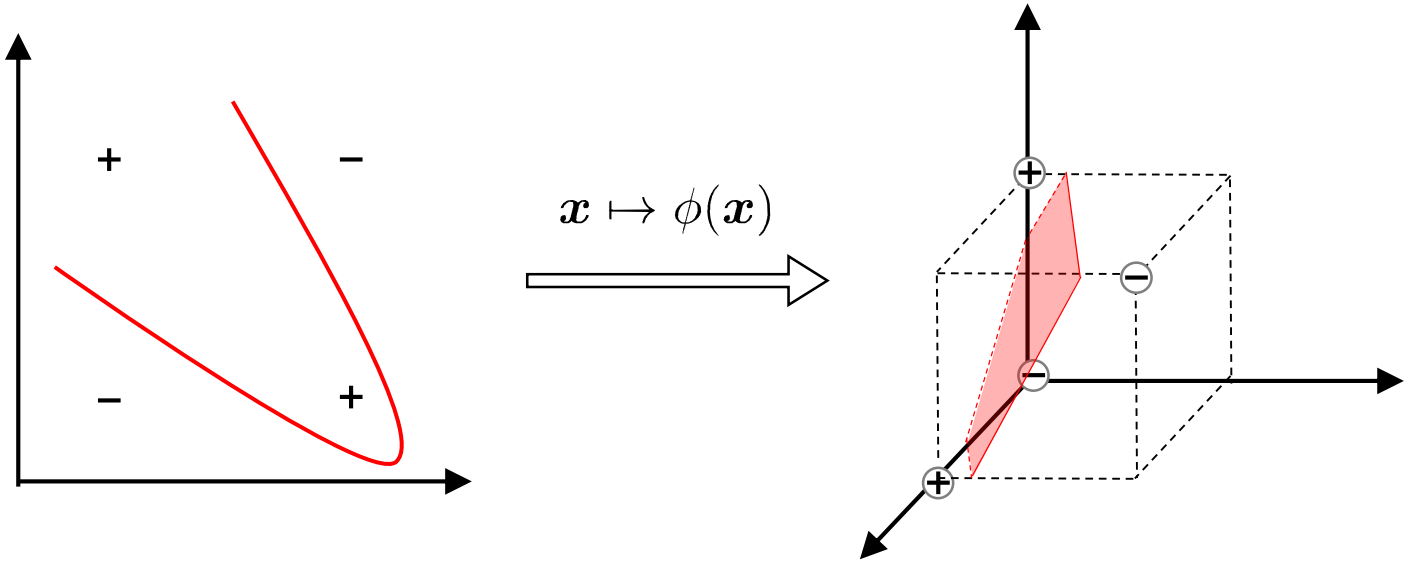
- 偏移项 b：通过支持向量来确定.

3. 核函数

线性不可分

-Q:若不存在一个能正确划分两类样本的超平面, 怎么办?

-A:将样本从原始空间映射到一个更高维的特征空间, 使得样本在这个特征空间内线性可分.



核支持向量机

- 设样本 $\boldsymbol{x}$ 映射后的向量为 $\phi(\boldsymbol{x})$，划分超平面为 $f(\boldsymbol{x}) = \boldsymbol{w}^\top \phi(\boldsymbol{x}) + b$

原始问题

$$\min_{\boldsymbol{w},b} \ \frac{1}{2}\|\boldsymbol{w}\|^2$$
$$\text{s.t.} \ \ y_i(\boldsymbol{w}^\top \phi(\boldsymbol{x}_i) + b) \geq 1, \ \ i = 1, 2, \ldots, m.$$

对偶问题

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \boxed{\phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j)} - \sum_{i=1}^{m}\alpha_i$$
$$\text{s.t.} \ \ \sum_{i=1}^{m}\alpha_i y_i = 0,$$

只以内积的形式出现

预测

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \phi(\boldsymbol{x}) + b = \sum_{i=1}^{m}\alpha_i y_i \boxed{\phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x})} + b$$

核函数

- 基本想法：不显式地设计核映射，而是设计核函数.

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^\top \phi(\boldsymbol{x}_j)$$

- Mercer定理(充分非必要)：只要一个对称函数所对应的核矩阵半正定，则它就能作为核函数来使用.

- 常用核函数：

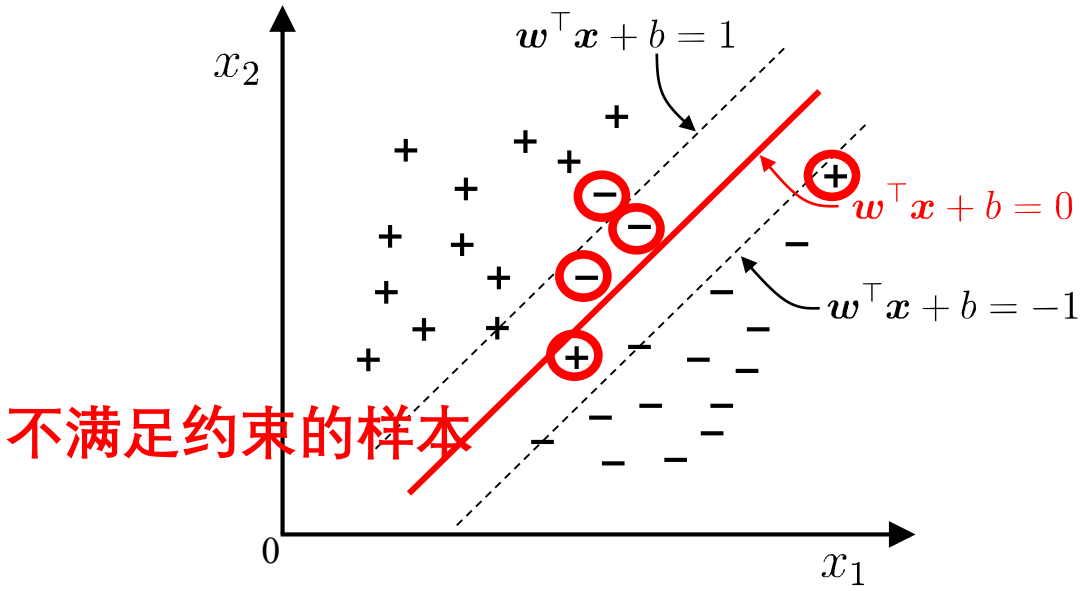
名称	表达式	参数
线性核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^\top \boldsymbol{x}_j$	
多项式核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i^\top \boldsymbol{x}_j)^d$	$d \geq 1$为多项式的次数
高斯核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\delta^2}\right)$	$\delta > 0$为高斯核的带宽(width)
拉普拉斯核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|}{\delta}\right)$	$\delta > 0$
Sigmoid核	$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \tanh(\beta \boldsymbol{x}_i^\top \boldsymbol{x}_j + \theta)$	tanh为双曲正切函数, $\beta > 0, \theta < 0$

4. 软间隔与正则化

软间隔

-Q:现实中, 很难确定合适的核函数使得训练样本在特征空间中线性可分; 一个线性可分的结果也很难断定是否是有过拟合造成的.

-A:引入"软间隔"的概念, 允许支持向量机在一些样本上不满足约束.



0/1损失函数

- 基本想法：最大化间隔的同时，让不满足约束的样本应尽可能少.

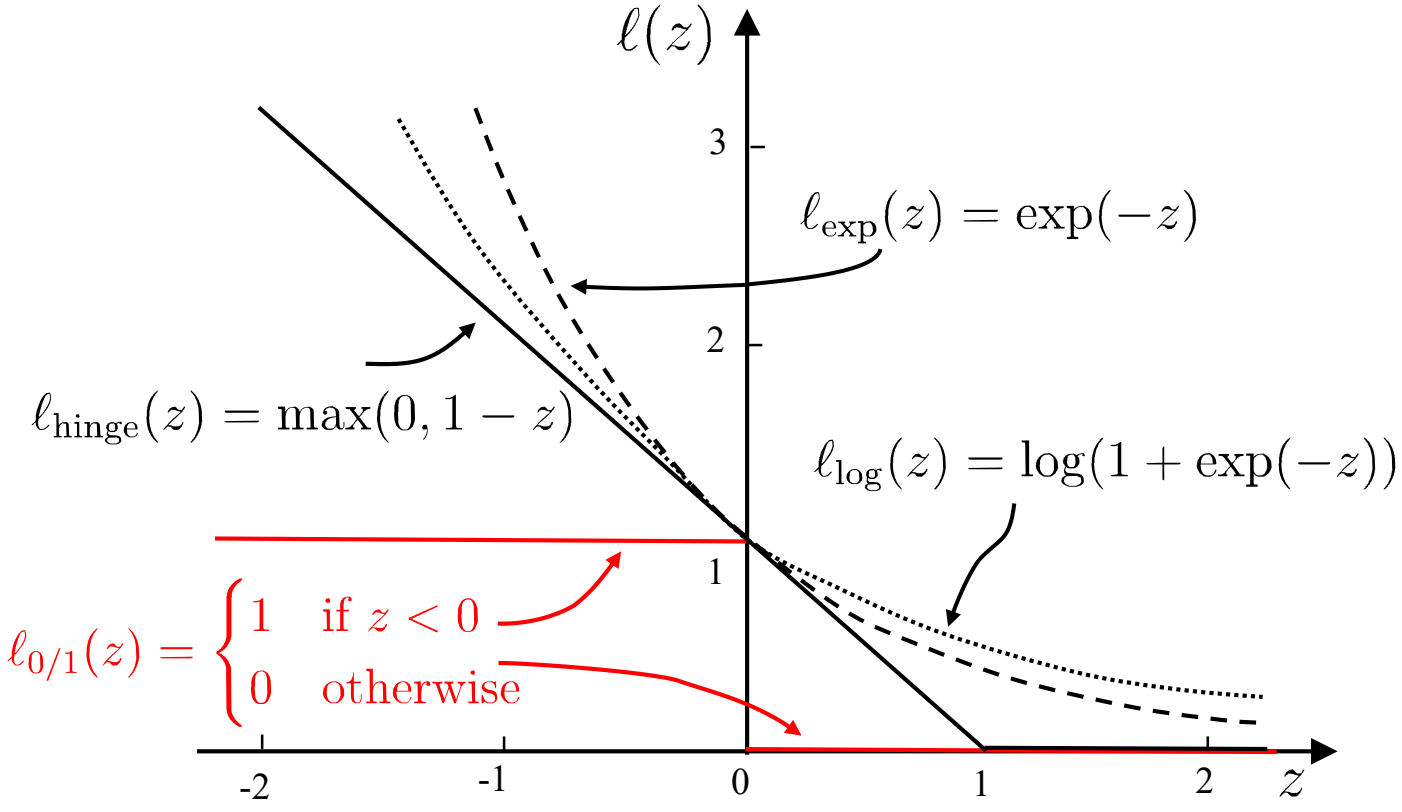
$$\min_{\boldsymbol{w},b} \ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m} l_{0/1}\left(y_i(\boldsymbol{w}^\top\phi(\boldsymbol{x}_i)+b)-1\right)$$

其中 $l_{0/1}$ 是"0/1损失函数"

$$l_{0/1} = \begin{cases} 1 & z < 0 \\ 0 & otherwise \end{cases}$$

- 存在的问题：0/1损失函数非凸、非连续，不易优化!

替代损失



替代损失函数数学性质较好，一般是0/1损失函数的上界

软间隔支持向量机

原始问题

$$\min_{\boldsymbol{w},b} \ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}\max\left(0, 1 - y_i(\boldsymbol{w}^\top\phi(\boldsymbol{x}_i) + b)\right)$$

对偶问题

$$\min_{\boldsymbol{\alpha}} \ \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j \phi(\boldsymbol{x}_i)^\top\phi(\boldsymbol{x}_j) - \sum_{i=1}^{m}\alpha_i$$

$$\text{s.t.} \ \sum_{i=1}^{m}\alpha_i y_i = 0, \ 0 \le \alpha_i \le C, \ i = 1, 2, \ldots, m.$$

根据KKT条件可推得最终模型仅与支持向量有关, 也即hinge损失函数依然保持了支持向量机解的稀疏性.

正则化

- 支持向量机学习模型的更一般形式

$$\min_f \ \Omega(f) + C\sum_{i=1}^{m} l(f(\boldsymbol{x}_i), y_i)$$

结构风险，描述模型的某些性质

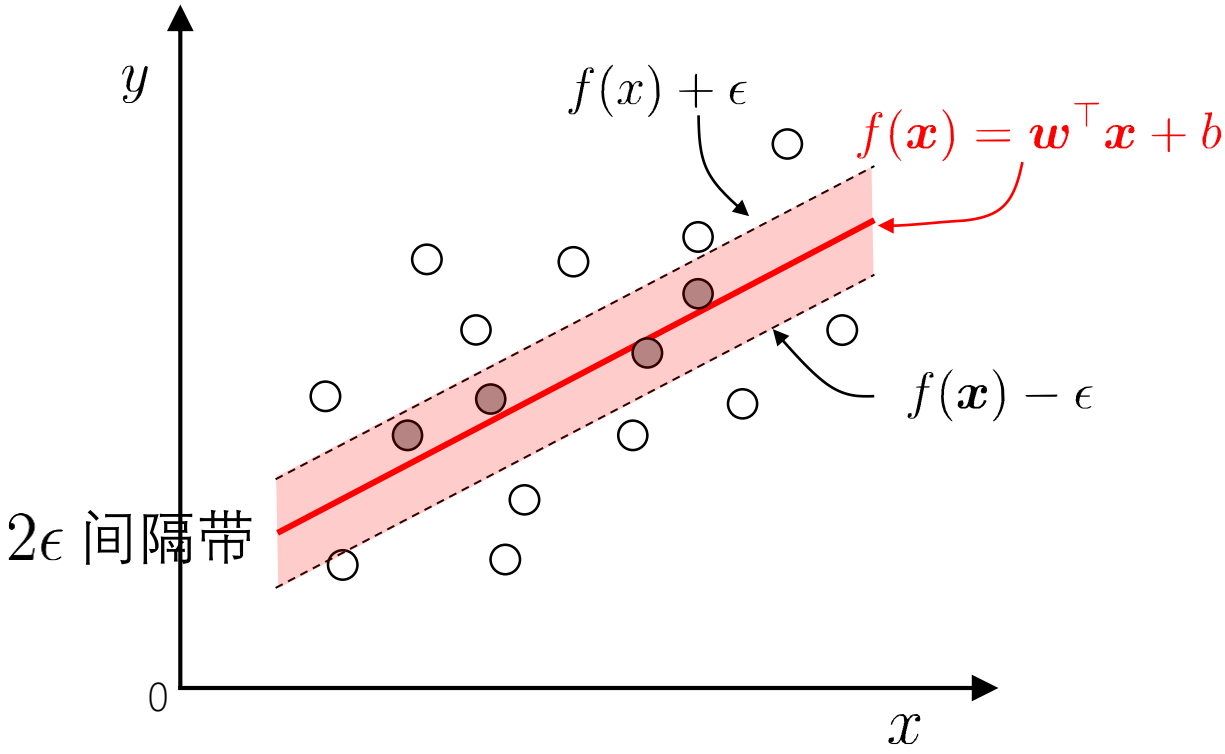
经验风险，描述模型与训练数据的契合程度

- 通过替换上面两个部分，可以得到许多其他学习模型
  - 逻辑回归(Logistic Regression)
  - ……

5. 支持向量回归

支持向量回归

特点: 允许模型输出和实际输出间存在 2ϵ 的偏差.



支持向量回归

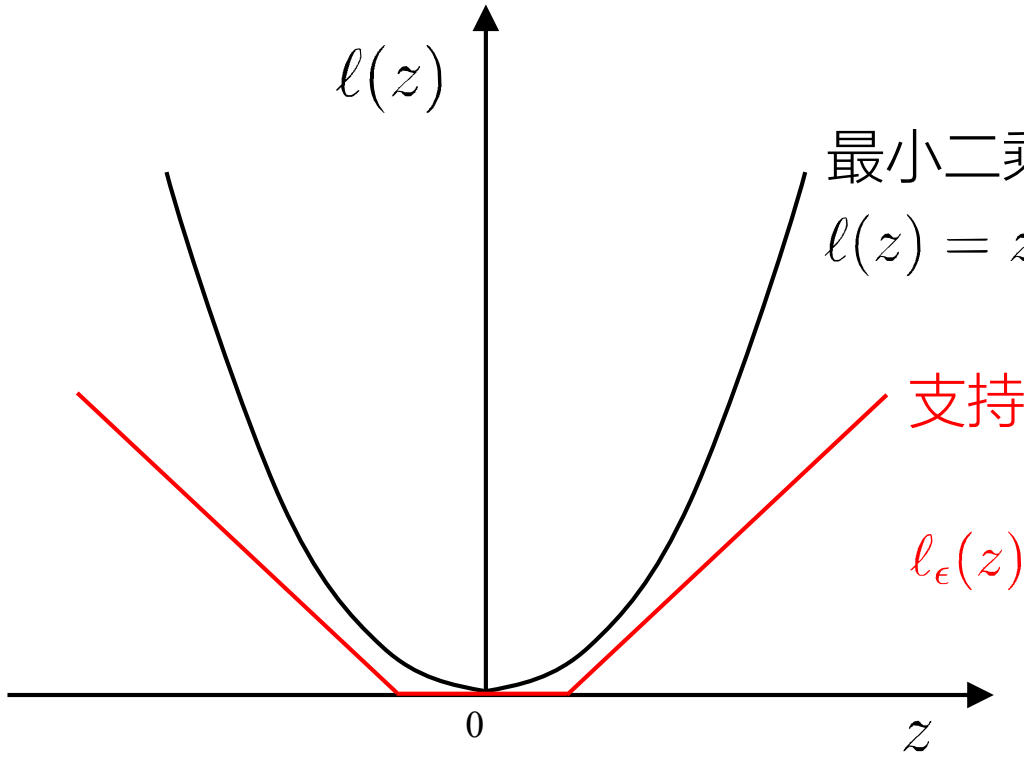
SVR的问题可以形式化为：

$$\min_{\boldsymbol{w},b} \ \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}\ell_\epsilon(f(\boldsymbol{x}_i) - y_i)$$

$$\ell_\epsilon(z) = \begin{cases} 0 & \text{if } |z| \leq \epsilon \\ |z| - \epsilon & \text{otherwise} \end{cases}$$

损失函数

落入中间 2ϵ 间隔带的样本不计算损失, 从而使得模型获得稀疏性.



最小二乘损失函数

$$\ell(z) = z^2$$

支持向量回归损失函数

$$\ell_\epsilon(z) = \begin{cases} 0 & \text{if } |z| \leq \epsilon \\ |z| - \epsilon & \text{otherwise} \end{cases}$$

形式化

引入两个松弛变量:

$$
\begin{aligned}
\min_{\boldsymbol{w},b,\xi_i,\hat{\xi}_i} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{m}(\xi_i + \hat{\xi}_i) \\
\text{s.t.} \quad & y_i - \boldsymbol{w}^\top \phi(\boldsymbol{x}_i) - b \leq \epsilon + \xi_i, \\
& y_i - \boldsymbol{w}^\top \phi(\boldsymbol{x}_i) - b \geq -\epsilon - \hat{\xi}_i, \\
& \xi_i \geq 0,\ \hat{\xi}_i \geq 0,\ i = 1, 2, \ldots, m.
\end{aligned}
$$

形式化

对偶问题

$$\min_{\boldsymbol{\alpha},\hat{\boldsymbol{\alpha}}} \ \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i-\hat{\alpha}_i)(\alpha_j-\hat{\alpha}_j)\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)+\sum_{i=1}^{m}(\alpha_i(\epsilon-y_i)+\hat{\alpha}_i(\epsilon+y_i))$$

$$\text{s.t.} \ \sum_{i=1}^{m}(\alpha_i-\hat{\alpha}_i)=0,$$

$$0\le\alpha_i\le C,\ 0\le\hat{\alpha}_i\le C.$$

预测

$$f(\boldsymbol{x})=\boldsymbol{w}^{\top}\phi(\boldsymbol{x})+b=\sum_{i=1}^{m}(\hat{\alpha}_i-\alpha_i)y_i\kappa(\boldsymbol{x}_i,\boldsymbol{x})+b$$

6. 核方法

表示定理

支持向量机 $$f(\boldsymbol{x}) = \boldsymbol{w}^\top \phi(\boldsymbol{x}) + b = \sum_{i=1}^{m} \alpha_i y_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}) + b$$

支持向量回归 $$f(\boldsymbol{x}) = \boldsymbol{w}^\top \phi(\boldsymbol{x}) + b = \sum_{i=1}^{m} (\hat{\alpha}_i - \alpha_i) y_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}) + b$$

结论: 无论是支持向量机还是支持向量回归, 学得的模型总可以表示成核函数的线性组合.

核线性判别分析

- 通过表示定理可以得到很多线性模型的"核化"版本
  - 核SVM
  - 核LDA
  - 核PCA
  - ……

Take Home Message

- 支持向量机的"最大间隔"思想
- 对偶问题及其解的稀疏性
- 通过向高维空间映射解决线性不可分的问题
- 引入"软间隔"缓解特征空间中线性不可分的问题
- 将支持向量的思想应用到回归问题上得到支持向量回归
- 将核方法推广到其他学习模型

成熟的SVM软件包

- LIBSVM
  http://www.csie.ntu.edu.tw/~cjlin/libsvm/
- LIBLINEAR
  http://www.csie.ntu.edu.tw/~cjlin/liblinear/
- SVM^{light} 、SVM^{perf} 、SVM^{struct}
  http://svmlight.joachims.org/svm_struct.html
- Pegasos
  http://www.cs.huji.ac.il/~shais/code/index.html

Thank you!

权小军 中山大学数据科学与计算机学院