



# 第1章 统计学习方法概论



# 主要内容

- 监督学习
- 统计的三要素
- 模型的评价和选择



# 1.1 监督学习

# 1. 样本数据

样本数据就是的 $(x, y)$ ，其中 $x$ 叫做输入数据(input data)， $y$ 叫做输出数据(output data)或者标签(label)/类别。通常 $x$ 和 $y$ 都是高维矩阵。

例： $k$ 个样本（样本也称为实例）构成的样本空间 $D$ 为：

$$D = \{ (x_1, y_1), (x_2, y_2), \dots, (x_k, y_k) \}$$

其中 $x_i$ 表示第 $i$ 个输入样本，若 $x_i$ 为 $d$ 维特征：

$$x_i = (x_i^1, x_i^2, \dots, x_i^d)$$

标签 $y$ 根据需求不同有各种形式：二值型，多值型和连续型

$x_1 = ( \text{形状} = \text{圆形} \quad \text{剥皮} = \text{难} \quad \text{味道} = \text{酸甜} ) , y_1 = \text{橙子}$   
 $x_2 = ( \text{形状} = \text{扁圆形} \quad \text{剥皮} = \text{易} \quad \text{味道} = \text{酸} ) , y_2 = \text{橘子}$   
 $x_3 = ( \text{形状} = \text{长圆形} \quad \text{剥皮} = \text{难} \quad \text{味道} = \text{甜} ) , y_3 = \text{橙子}$   
...



**输入与输出的映射模型：**

**$F = ( ( \text{形状} = * \quad \text{剥皮} = \text{难} \quad \text{味道} = * ) , \text{橙子} )$**

## 2. 样本分布

分布(distribution)：样本空间的全体样本服从的一种规律

**独立同分布**(independent and identically distributed):

随机变量 $X_1$ 和 $X_2$ 独立是指 $X_1$ 的取值不影响 $X_2$ 的取值， $X_2$ 的取值也不影响 $X_1$ 的取值。

随机变量 $X_1$ 和 $X_2$ 同分布是指 $X_1$ 和 $X_2$ 具有相同的分布形状和相同的分布参数。

### 3. 数据集

对于机器学习而言，**不是**所有的数据集 $D$ ，都用于训练学习模型，而是会被分为两个部分：训练数据和测试数据。

**训练数据**(training data):训练数据用于训练**学习模型**，通常比例不低于总数据量的一半

**测试数据**(testing data)：用于衡量**学习模型**的性能好坏。

## 4、 监督学习

机器学习分为监督学习、无监督学习、半监督和强化学习

**监督学习**(supervised learning)：从**标签的训练数据**来推断的机器学习任务。

在监督学习中，每个实例都是由一个输入数据和一个期望的输出值组成。监督学习算法是分析该训练数据，并**产生一个推断的功能**，其可以用于映射出新的实例。采用一个最佳的算法模型来决定未知实例的标签/类别。

**无监督学习**(unsupervised learning)：按照样本的**性质**把它们**自动地分成很多组**，每组数据具有类似性质

输入数据没有标签，需要根据样本间的相似性对样本进行分类(聚类，clustering)试图使类内差距最小化，类间差距最大化





## 半监督学习：


在训练阶段结合大量未标记的数据和少量标签数据。与使用所有标签数据的模型相比，使用训练集的训练模型在训练时可以更为准确，而且训练成本更低。

## 强化学习：

智能系统从环境到行为映射的学习，以使奖励信号(强化信号)函数值最大。如果Agent的某个行为策略导致环境正的奖赏(强化信号)，那么Agent以后产生这个行为策略的趋势便会加强。

## 5. 机器学习解决问题

- Ø 分类 ( 监督学习 )
- Ø 回归 ( 监督学习 )
- Ø 聚类 ( 无监督学习 )
- Ø 关联 ( 无监督学习 )



分类(classification)：预测值是离散值


比如把人分为好人和坏人之类的学习任务

二分类(binary classification)： 只涉及两个类别的分类任务

正例(positive class)、反例 (negative class)

多分类 ( multi-class classification )

涉及多个类别的分类



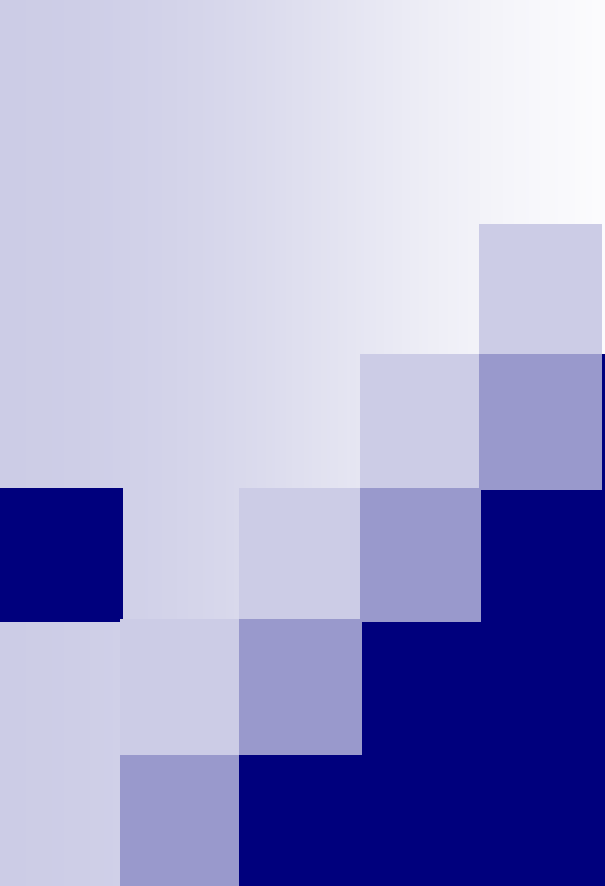
回归（ regression ）：预测值是连续值。

比如你的好人程度达到了0.9，0.6之类

聚类（ clustering ）：把训练集中的对象分为若干组

关联（ association rule ）：用来发现事情之间的联系。

最早是为了发现超市交易数据库中不同的商品之间的关系。



## 1.2 学习三要素

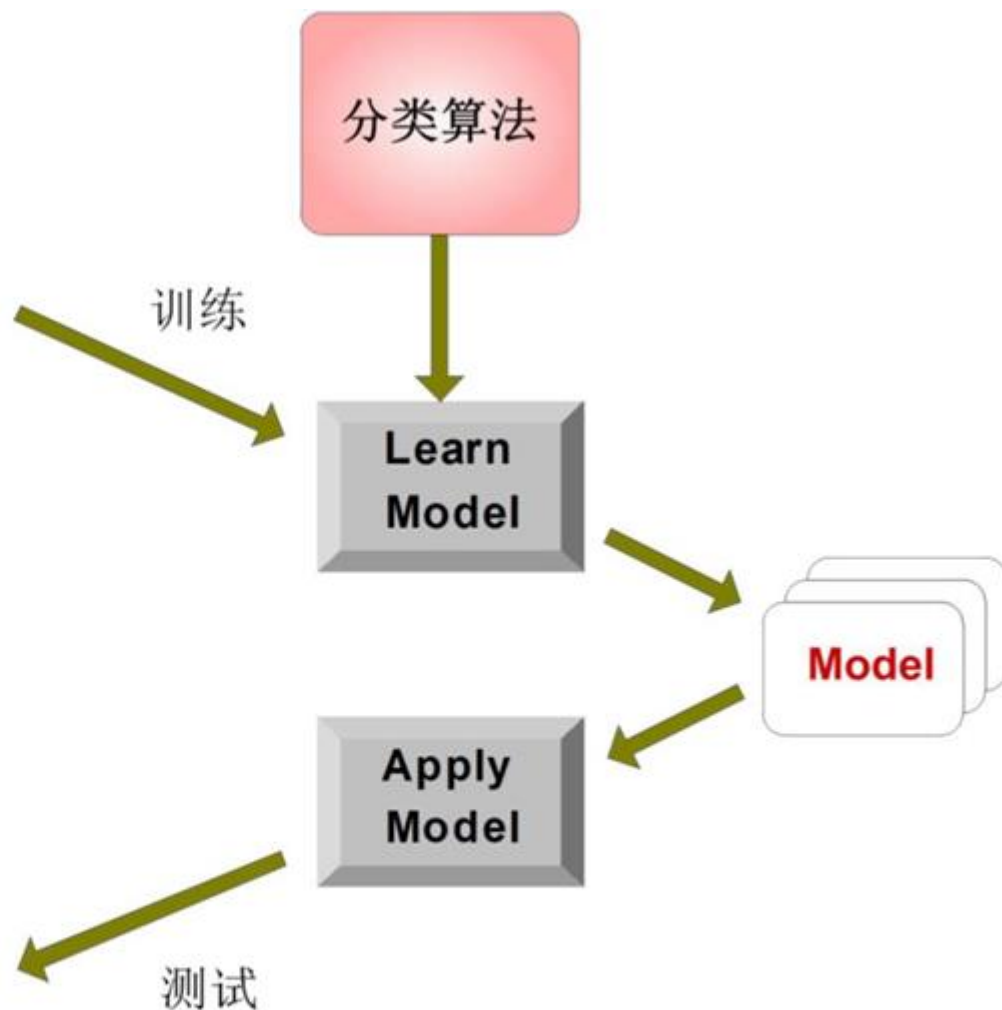
# 机器学习的目的

	属性 1	属性 2	属性 3	分类
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

训练集合

	属性 1	属性 2	属性 3	分类
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

测试集合



方法=模型+策略+算法

# 1、模型

模型是输入到输出的映射。模型的集合，称为**假设空间**。

当假设空间F为**决策函数的集合**： $F = \{f | Y = f(X)\}$

F实质为参数向量决定的函数族： $F = \{f | Y = f_{\theta}(X), \theta \in R^n\}$

当假设空间F为**条件概率的集合**： $F = \{P | P(Y|X)\}$

F实质是参数向量决定的条件概率分布族 $F = \{P | P_{\theta}(Y|X), \theta \in R^n\}$

## 2、策略

**损失函数**：度量模型一次预测的好坏。

0-1损失函数0-1 loss function

$$L(Y, f(X)) = \begin{cases} 1 & Y \neq f(X) \\ 0 & Y = f(X) \end{cases}$$

平方损失函数quadratic loss function

$$L(Y, f(X)) = (Y - f(X))^2$$

绝对损失函数absolute loss function

$$L(Y, f(X)) = |Y - f(X)|$$

对数损失函数logarithmic loss function

或对数似然损失函数loglikelihood loss function

$$L(Y, P(Y|X)) = -\log P(Y|X)$$



**风险函数**：度量平均意义上模型预测的好坏。

损失函数期望:

$$R_{\text{exp}}(f) = E_p[L(Y, f(X))] = \iint_{x \times y} L(y, f(x))P(x, y)dxdy$$

风险函数risk function 期望损失expected loss

对于给定训练数据集

$$T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

经验风险empirical risk , 经验损失empirical loss

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

# 经验风险最小化与结构风险最小化

经验风险最小化最优模型

$$\min_{f \in F} \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i))$$

当样本容量很小时，经验风险最小化学习的效果未必很好，会产生“过拟合over-fitting”

# 经验风险最小化与结构风险最小化

为防止过拟合提出的策略，结构风险最小化structure risk minimization，等价于正则化（regularization），加入正则化项regularizer，或罚项penalty term：

$$R_{srmm}(f) = \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i)) + \lambda J(f)$$

$J(f)$ 为模型复杂度，模型 $f$ 越复杂，复杂度 $J(f)$ 越大， $\lambda \geq 0$ 是惩罚系数

### 3、算法

求最优模型就是求解最优化问题：

$$\min_{f \in F} \left( \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i)) + \lambda J(f) \right)$$

难点：

全局最优  
算法高效



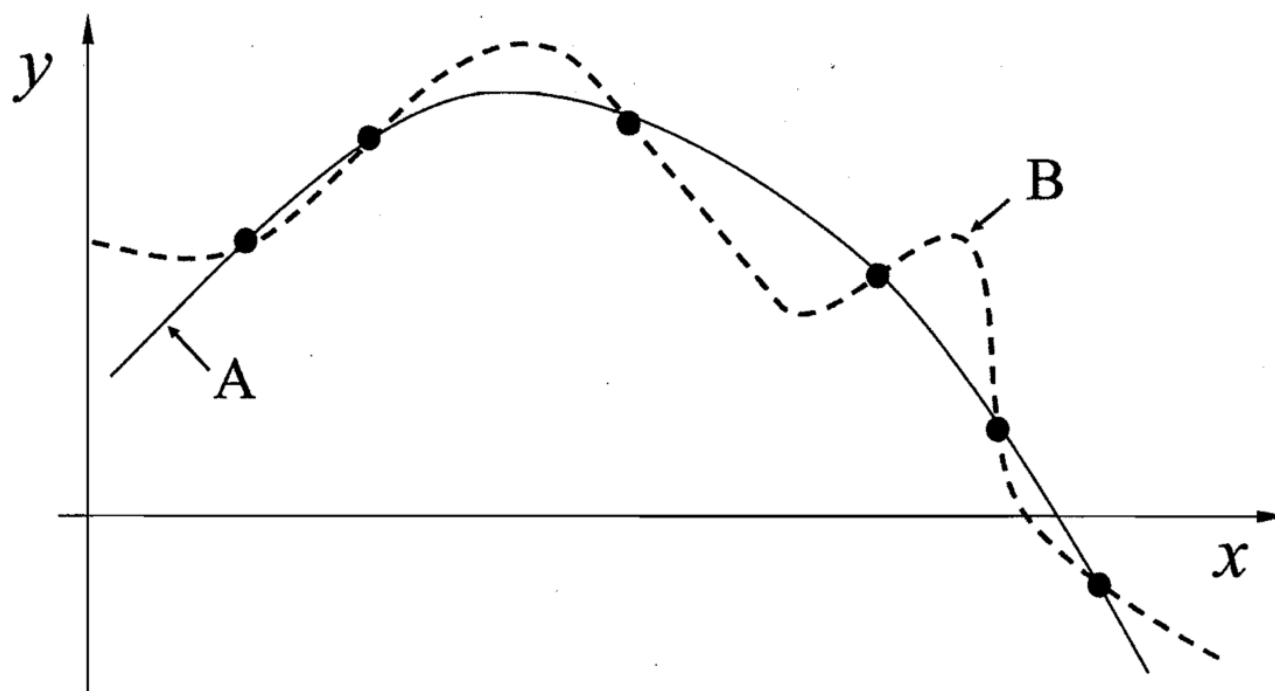
## 1.3 模型评估与模型选择

# 主要内容

- 奥卡姆剃刀定理
- 训练误差和测试误差
- 过拟合
- 正则化
- 泛化能力
- 生成模型和判决模型
- 评估方法
- 性能度量

# 1、Occam's razor (奥卡姆剃刀定理)

原理称为“如无必要，勿增实体”



## 2、训练误差和测试误差

训练误差：训练数据集的平均损失率

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

测试误差：测试数据集的平均损失率

$$e_{test}(\hat{f}) = \frac{1}{N'} \sum_{i=1}^{N'} L(y_i, \hat{f}(x_i))$$



损失函数是0-1损失时，测试误差：

$$e_{test}(\hat{f}) = \frac{1}{N'} \sum_{i=1}^{N'} I(y_i \neq \hat{f}(x_i))$$

测试数据集的准确率：

$$r_{test}(\hat{f}) = \frac{1}{N'} \sum_{i=1}^{N'} I(y_i = \hat{f}(x_i))$$

### 3、过拟合

当假设空间含有不同的复杂度（如不同参数个数）的模型时，我们选择的模型应该逼近“真实”的模型。

学习时，选择的模型所包含的参数过多，以至于出现对已知数据预测得很好，但对未知数据预测得很差的现象，称为**过拟合**。

# 过拟合与模型选择 - 【多项式曲线拟合】

【例1.1】假设给定训练数据集 $T=\{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ ,  $x_i \in \mathbb{R}$ 是输入 $x$ 的观测值,  $y_i \in \mathbb{R}$ 是响应的输出观测值, 用多项式函数曲线拟合数据。

$$f_M(X, W) = W_0 + W_1 X + \dots + W_M X^M = \sum_{j=0}^M W_j X^j$$

经验风险最小：

$$L(W) = \frac{1}{2} \sum_{i=1}^N (f(X_i, W) - y_i)^2$$

代入 $f_M(x, w)$

$$L(W) = \frac{1}{2} \sum_{i=1}^N \left( \sum_{j=0}^M W_j X_i^j - y_i \right)^2$$

上式取值最小，则其关于 $w_k$ 求偏导，并令偏导=0，则

$$\frac{\partial S}{\partial w_0} = \sum_{i=1}^N 2[f(x_i) - y_i] = 0 \Rightarrow \sum_{i=1}^N [f(x_i) - y_i] = 0 \Rightarrow \sum_{i=1}^N f(x_i) = \sum_{i=1}^N y_i$$

$$\frac{\partial S}{\partial w_1} = \sum_{i=1}^N 2x_i[f(x_i) - y_i] = 0 \Rightarrow \sum_{i=1}^N x_i[f(x_i) - y_i] = 0 \Rightarrow \sum_{i=1}^N x_i f(x_i) = \sum_{i=1}^N x_i y_i$$

$$\frac{\partial S}{\partial w_m} = \sum_{i=1}^N 2nx_i^m[f(x_i) - y_i] = 0 \Rightarrow \sum_{i=1}^N x_i^m[f(x_i) - y_i] = 0 \Rightarrow \sum_{i=1}^N x_i^m f(x_i) = \sum_{i=1}^N x_i^m y_i$$

## 将上面各等式写成方程组的形式

$$\sum_{i=1}^N f(x_i) = \sum_{i=1}^N y_i \Rightarrow$$


$$a_0 N + a_1 \sum_{i=1}^N x_i + a_2 \sum_{i=1}^N x_i^2 + \cdots + a_m \sum_{i=1}^N x_i^n = \sum_{i=1}^N y_i$$

$$\sum_{i=1}^N x_i f(x_i) = \sum_{i=1}^N x_i y_i \Rightarrow$$

$$a_0 x_i + a_1 \sum_{i=1}^N x_i^2 + a_2 \sum_{i=1}^N x_i^3 + \cdots + a_m \sum_{i=1}^N x_i^{m+1} = \sum_{i=1}^N x_i y_i$$

$$\sum_{i=1}^N x_i^m f(x_i) = \sum_{i=1}^N x_i^m y_i \Rightarrow$$

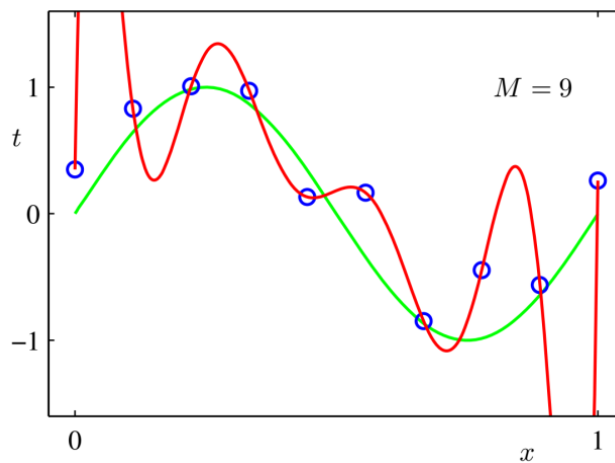
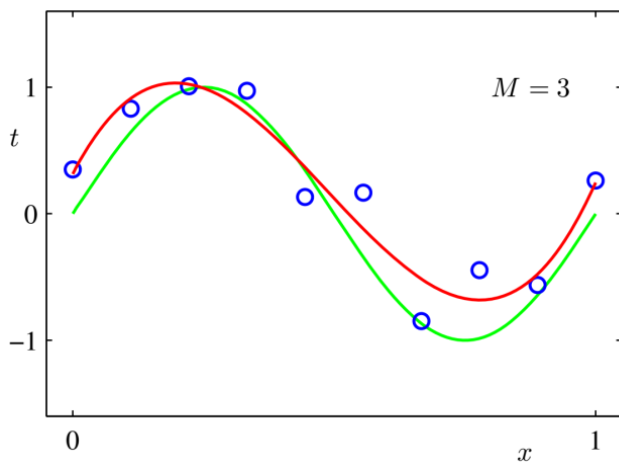
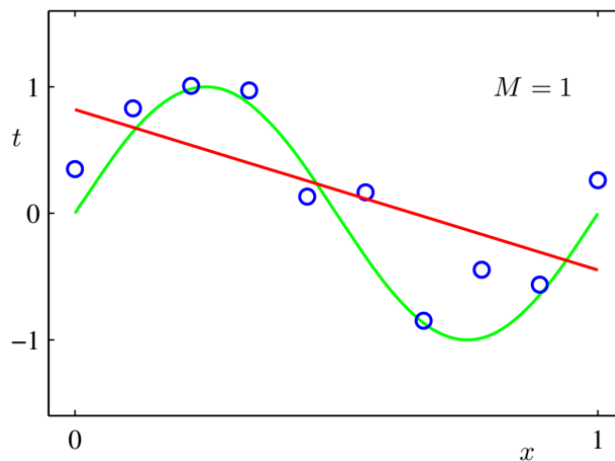
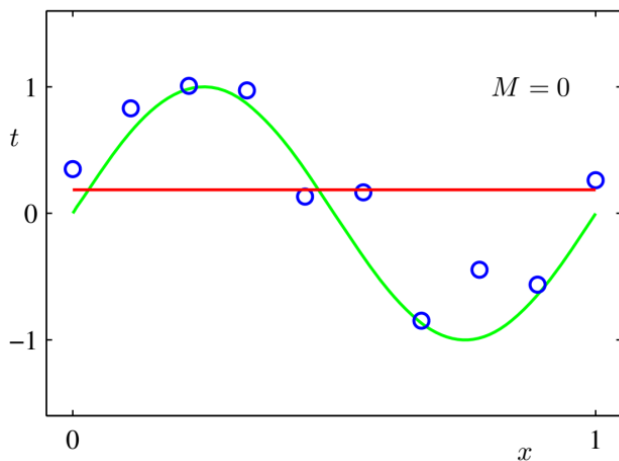
$$a_0 x_i^n + a_1 \sum_{i=1}^N x_i^{n+1} + a_2 \sum_{i=1}^N x_i^{n+2} + \cdots + a_m \sum_{i=1}^N x_i^{2m} = \sum_{i=1}^N x_i^m y_i$$



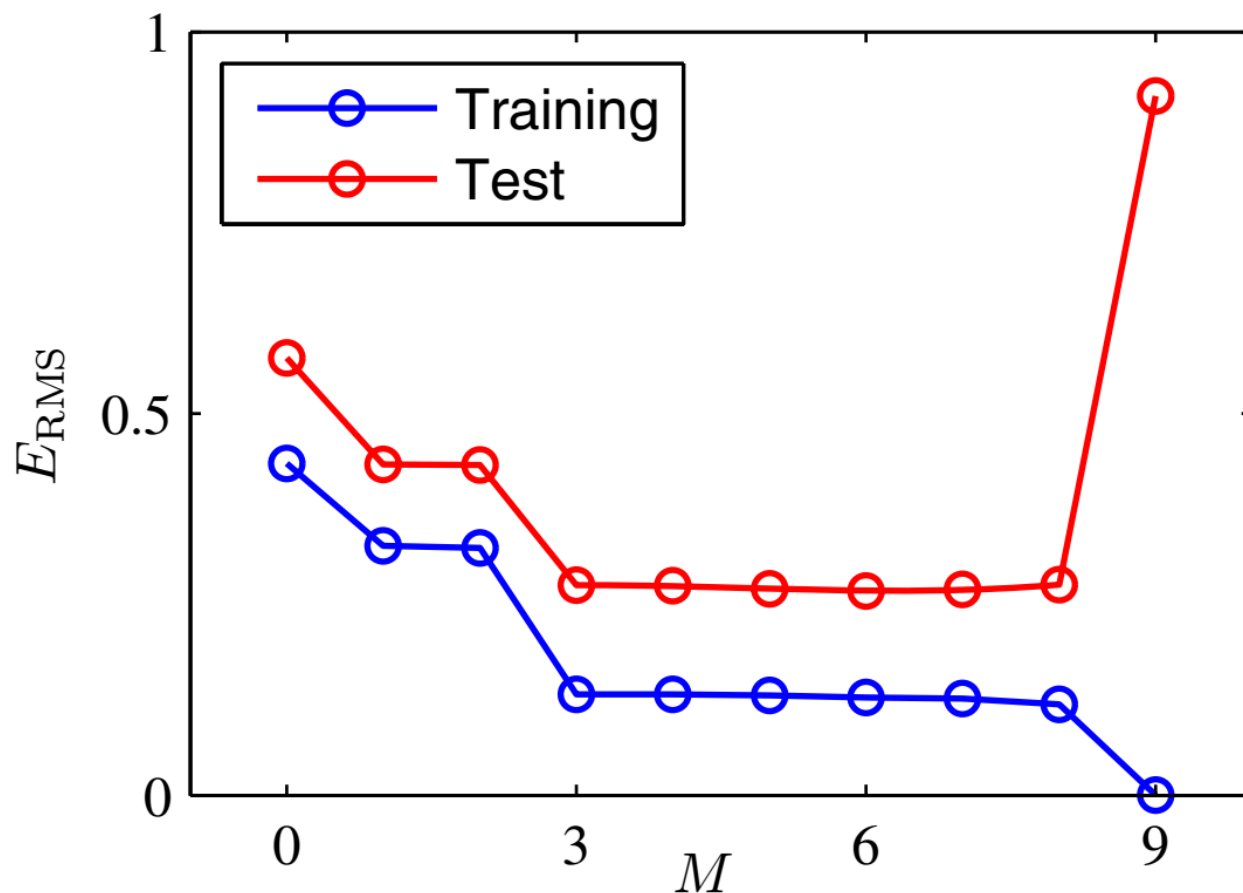
$$\begin{pmatrix}
 m & \sum_{i=1}^N X_i & \cdots & \sum_{i=1}^N X_i^k & \cdots & \sum_{i=1}^N X_i^m \\
 \sum_{i=1}^N X_i & \sum_{i=1}^N X_i^2 & \cdots & \sum_{i=1}^N X_i^{k+1} & \cdots & \sum_{i=1}^N X_i^{m+1} \\
 \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
 \sum_{i=1}^N X_i^k & \sum_{i=1}^N X_i^{k+1} & \cdots & \sum_{i=1}^N X_i^{2k} & \cdots & \sum_{i=1}^N X_i^{m+k} \\
 \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
 \sum_{i=1}^N X_i^m & \sum_{i=1}^N X_i^{n+1} & \cdots & \sum_{i=1}^N X_i^{n+k} & \cdots & \sum_{i=1}^N X_i^{2m}
 \end{pmatrix} \cdot \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_k \\ \vdots \\ w_m \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^N y_i \\ \sum_{i=1}^N X_i y_i \\ \vdots \\ \sum_{i=1}^N X_i^k y_i \\ \vdots \\ \sum_{i=1}^N X_i^m y_i \end{pmatrix}$$

克莱姆法则： $w_j = D_j / D$ ， $D_j$ 是将 $b$ 替换第 $j$ 列的行列式

练习：有样本数据(1,2),(3,4),(5,6),(7,8),(9,10),(11,12),(13,14),(15,16),(17,18)，利用多项式 $f(x) = w_0 + w_1x + w_2x^2$ 拟合曲线，以及python编程求出 $w_0, w_1, w_2$ 。



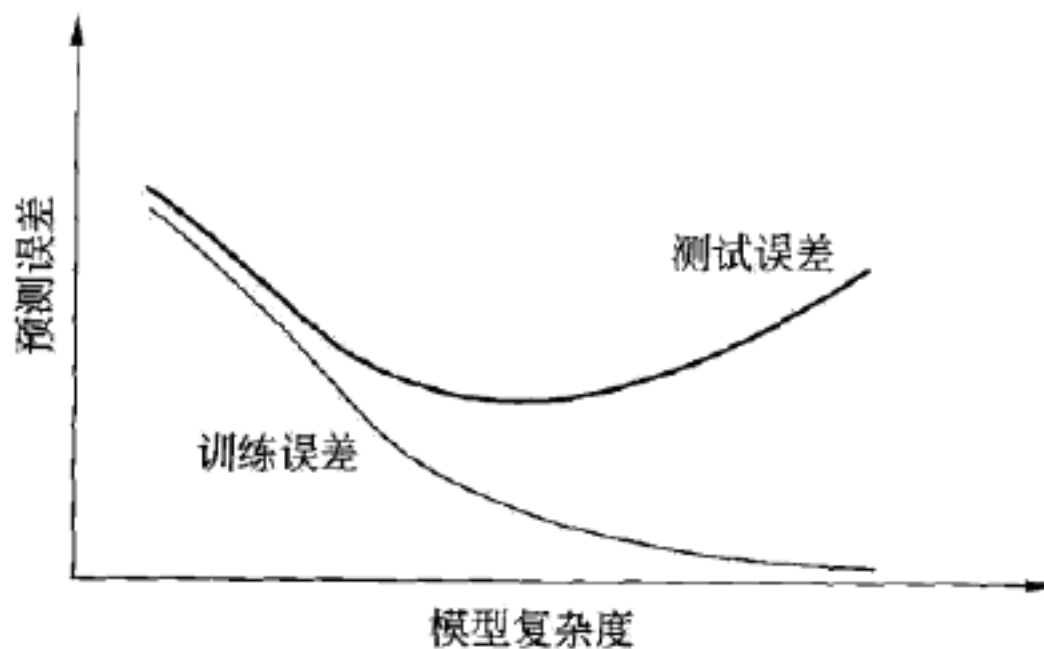
绿色： $\sin(2\pi x)$ 加入噪声产生的样本点  
红色：拟合曲线



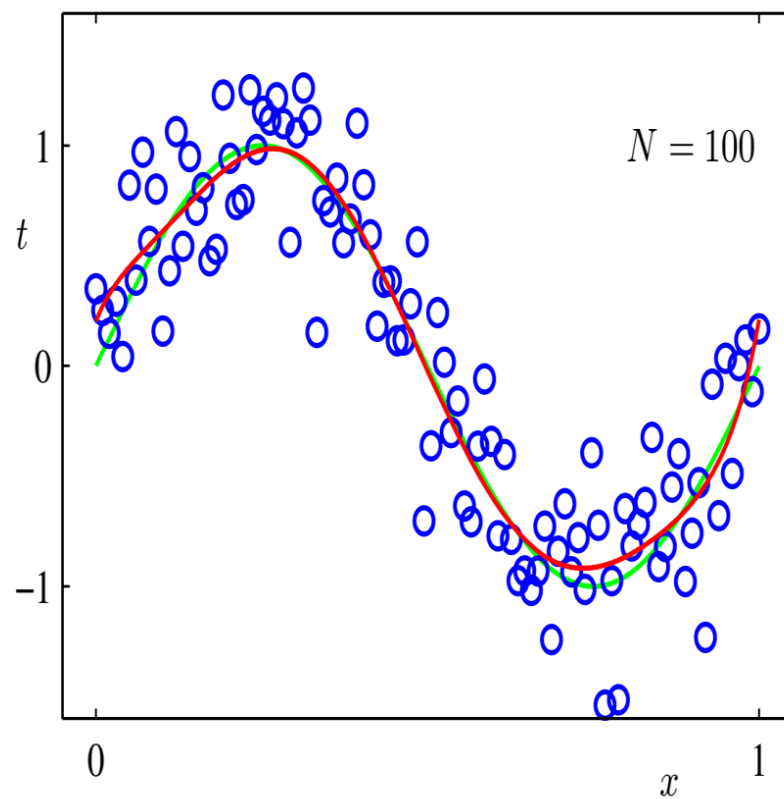
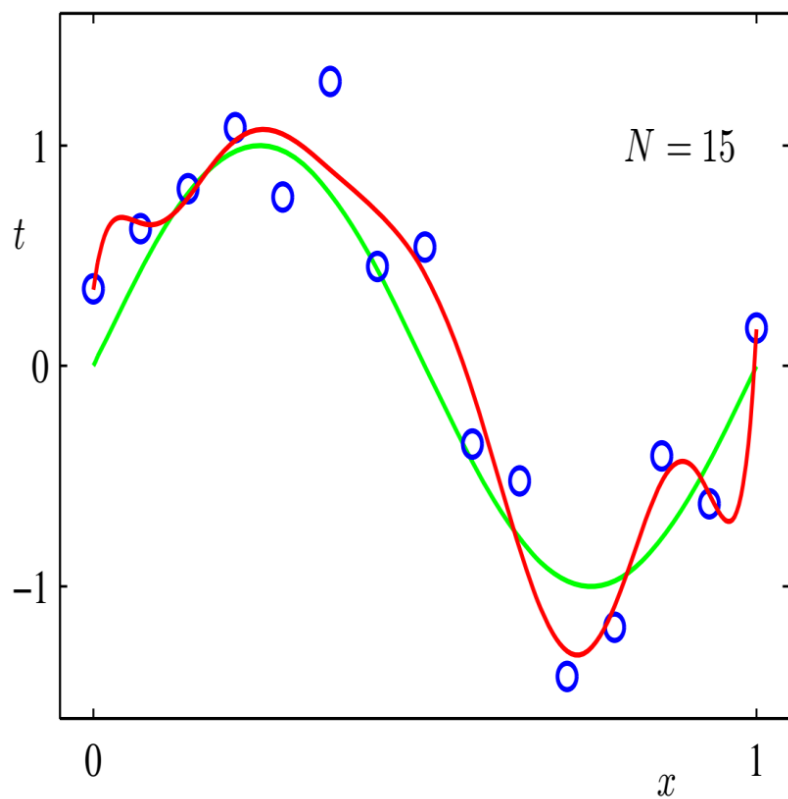
随着多项式次数（模型复杂度）增加，训练误差减小，但测试误差是先减小，后增加。



# 模型复杂度和误差之间的关系



增加训练样本的数量，可以防止过拟合。



## 4、正则化

一般形式：

$$\min_{f \in F} \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i)) + \lambda J(f)$$

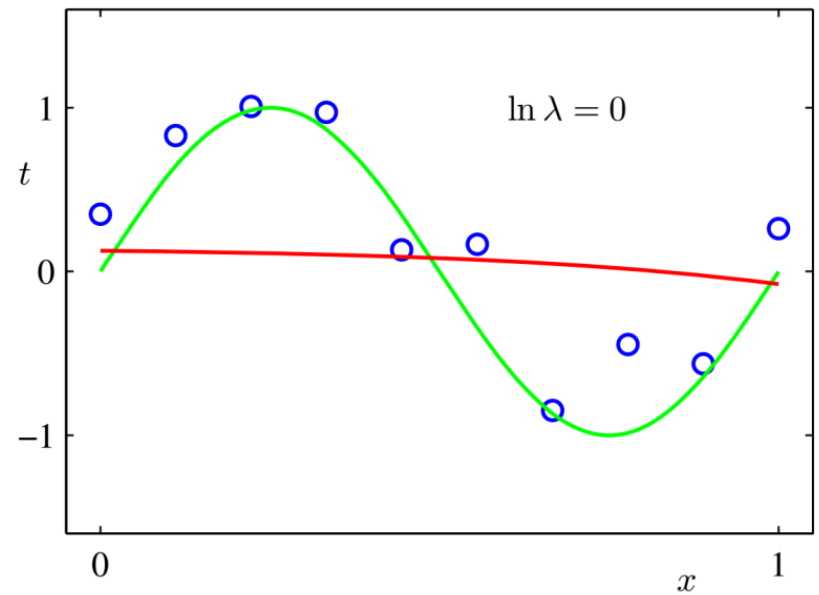
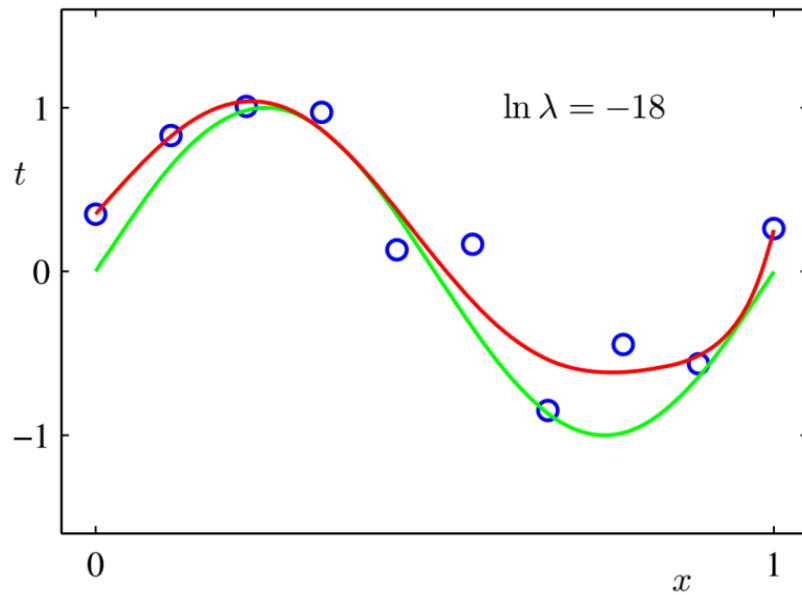
回归问题

$$L(w) = \frac{1}{N} \sum_{i=1}^N (f(x_i) - y_i)^2 + \frac{\lambda}{2} \|w\|^2$$

$$L(w) = \frac{1}{N} \sum_{i=1}^N (f(x_i, w) - y_i)^2 + \frac{\lambda}{2} \|w\|_1$$

范数 $\|w\|$  :  $L_p = \sqrt[p]{\sum_{i=1}^n x_i^p}$ ,  $x = (x_1, x_2, \dots, x_n)$

$$L(w) = \frac{1}{N} \sum_{i=1}^N \{f_M(x_i, w) - y_i\}^2 + \frac{\lambda}{2} \|w\|^2$$



	$M = 0$	$M = 1$	$M = 6$	$M = 9$		$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
$w_0^*$	0.19	0.82	0.31	0.35	$w_0^*$	0.35	0.35	0.13
$w_1^*$		-1.27	7.99	232.37	$w_1^*$	232.37	4.74	-0.05
$w_2^*$			-25.43	-5321.83	$w_2^*$	-5321.83	-0.77	-0.06
$w_3^*$			17.37	48568.31	$w_3^*$	48568.31	-31.97	-0.05
$w_4^*$				-231639.30	$w_4^*$	-231639.30	-3.89	-0.03
$w_5^*$				640042.26	$w_5^*$	640042.26	55.28	-0.02
$w_6^*$				-1061800.52	$w_6^*$	-1061800.52	41.32	-0.01
$w_7^*$				1042400.18	$w_7^*$	1042400.18	-45.95	-0.00
$w_8^*$				-557682.99	$w_8^*$	-557682.99	-91.53	0.00
$w_9^*$				125201.43	$w_9^*$	125201.43	72.68	0.01

$\lambda$ 增加，模型参数值减小，抑制模型的过拟合现象

## 5、泛化能力generalization ability

泛化能力是由该方法学习到的模型对未知数据的预测能力

泛化误差：学习模型对未知数据的误差。

$$R_{\text{exp}}(\hat{f}) = E_P[L(Y, \hat{f}(X))] = \int_{X \times Y} L(y, \hat{f}(x)) P(x, y) dx dy$$

## 6、生成模型和判决模型

监督学习的目的就是学习一个模型，模型的一般形式：

决策函数： $Y=f(X)$


或者条件概率分布： $P(Y|X)$

生成方法Generative approach由数据学习联合概率分布 $P(X,Y)$ ，然后求出条件概率分布 $P(Y|X)$ 作为预测模型，即生成模型：

$$P(Y \mid X) = \frac{P(X, Y)}{P(X)}$$

朴素贝叶斯法和隐马尔科夫模型

生成方法Generative approach 对应生成模型generative model



**判别方法**discriminative approach直接学习决策函数 $f(X)$ 或者条件概率分布 $P(Y|X)$ 作为预测模型。即给定输入 $X$ ，应该预测什么样的输出 $Y$ 。

- K近邻，感知机，决策树，logistic 回归等

判别方法discriminative approach对应判别模型  
discriminative model



## 二者各有优缺点

### 生成模型：

- 还原联合概率，而判别模型不能；
- 学习收敛速度快，当样本容量增加时，学到的模型可以更快收敛
- 当存在**隐变量**时，可以使用生成模型，而判别模型不行。

### 判别模型：

- 直接学习决策函数或条件概率，学习的准确率更高；
- 可以对数据进行抽象，定义特征和使用特征，可以简化学习问题

## 7、模型评估方法

### (1)留出法 Hold-out

$$D = S \cup T$$

$$S \cap T = \emptyset$$

( S训练集 , T测试集 )

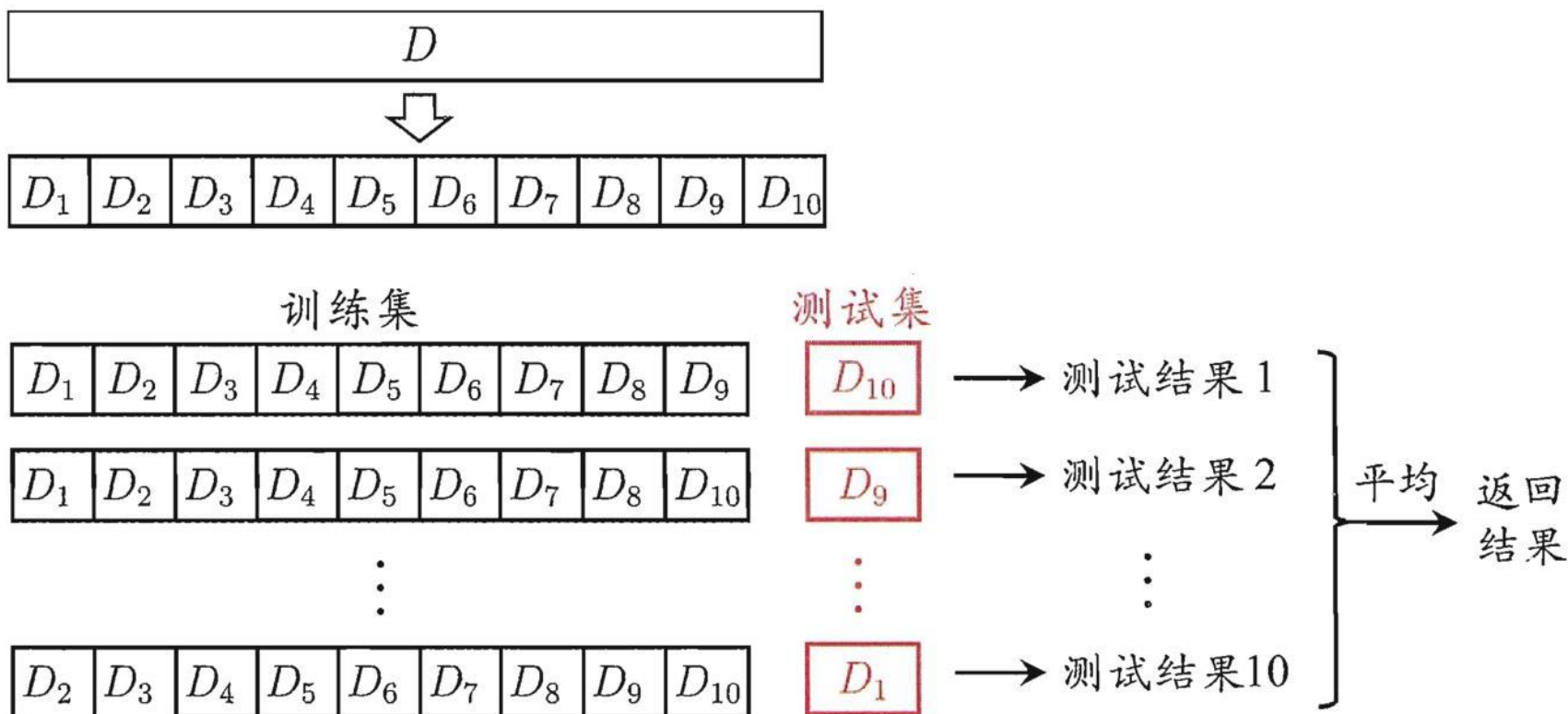
#### 注意点：

- 训练/测试集的划分尽可能保持数据分布的一致性，避免引入额外偏差；
- 存在多种划分方式对初始数据集进行分割，采用若干次随机划分，重复实验。
- 通常训练集：2/3-4/5样本点

## (2)交叉验证法cross validation

$D \rightarrow k$ 个大小相等的互斥子集,  $D = D_1 \cup D_2 \cup \dots \cup D_k$ ,  $D_i \cap D_j = \emptyset$

$K-1$ 个子集并集为训练集, 1个测试集



### (3)自助法 bootstrapping

自助采样法：m个样本的数据集D，进行采样产生数据集D'，每次采样一个样本放入D'。重复执行m次。

M次采样中始终不被采样的概率：

$$\lim_{m \rightarrow \infty} \left(1 - \frac{1}{m}\right)^m = \frac{1}{e} = 0.368$$

测试集：D \ D'

优点：

适用于数据集较小，难以有效划分训练和测试集的情况；  
从数据集产生不同的训练集，适用于集成学习方法；

缺点：

产生的训练集改变了初始数据集的分布，会引入估计偏差

## 8、性能度量

### (1) 错误率和精度

回归任务-均方误差：

$$E(f; D) = \frac{1}{m} \sum_{i=1}^m (f(x_i) - y_i)^2$$

二分类任务的错误率和精度

$$E(f; D) = \frac{1}{m} \sum_{i=1}^m I(f(x_i) \neq y_i)$$

$$acc(f; D) = \frac{1}{m} \sum_{i=1}^m I(f(x_i) = y_i)$$

## ( 2 ) 查准率precision 、查全率recall与FI

### 二分类-混淆矩阵

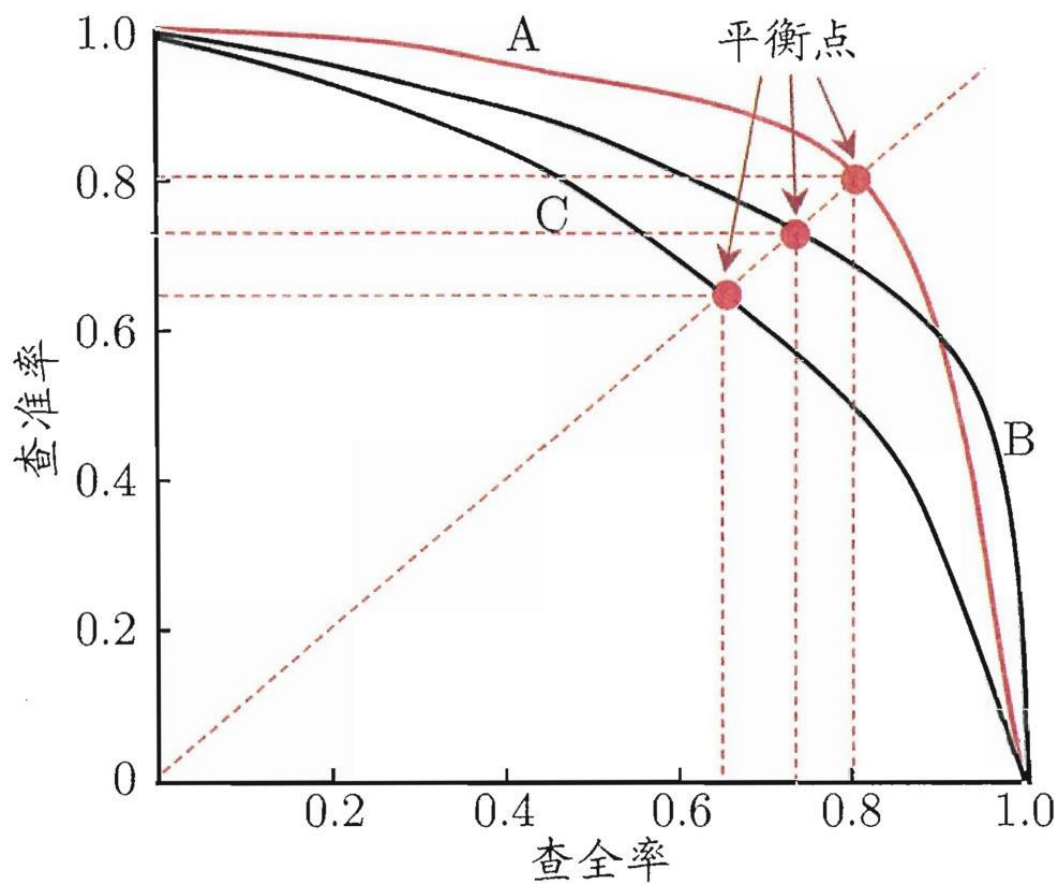
真实情况	预测结果	
	正例	反例
正例	$TP$ (真正例)	$FN$ (假反例)
反例	$FP$ (假正例)	$TN$ (真反例)

查准率(准确率) : 
$$P = \frac{TP}{TP + FP}$$

查全率(召回率) : 
$$R = \frac{TP}{TP + FN}$$

# PR曲线

根据模型的预测输出结果（一般为一个实值或概率）对测试样本进行排列，排在最前面的是模型认为最有可能是正例的样本，排在最后的是模型认为“最不可能”是正例的样本，按此顺序逐个把样本作为正例进行预测，每次计算出查全率和查准率



$A > B > C$

# ROC(Receiver Operating Characteristic)

## AUC(Area Under ROC Curve)

学习器对测试样本的评估结果一般为一个实值或概率，设定一个阈值，大于阈值为正例，小于阈值为负例，因此这个实值的好坏直接决定了学习器的泛化性能，若将这些实值排序，则排序的好坏决定了学习器的性能高低。

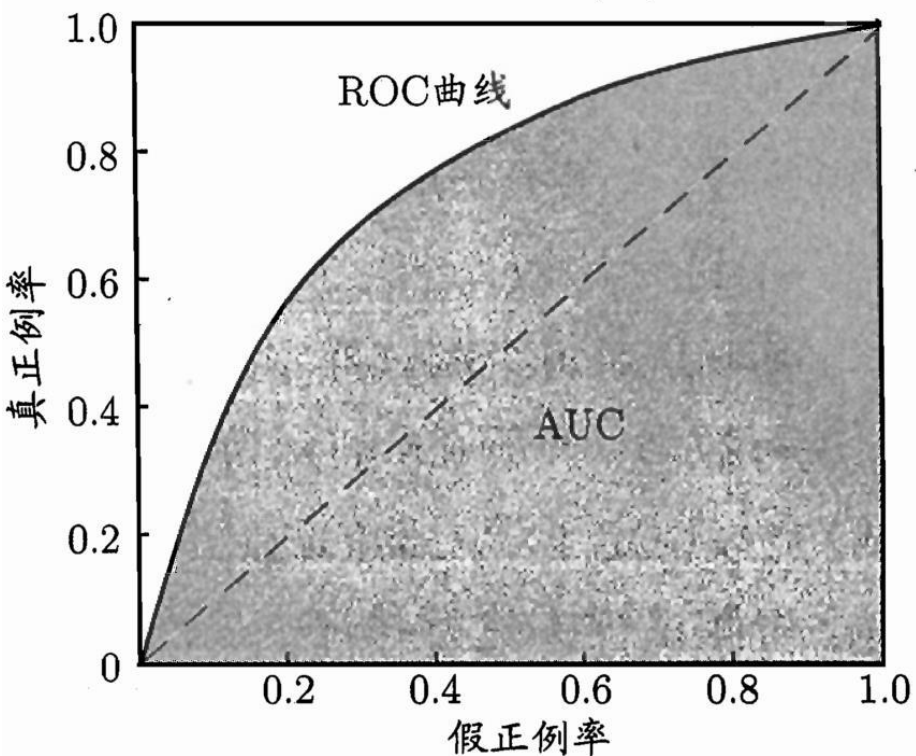
ROC曲线正是从这个角度出发来研究学习器的泛化性能，ROC曲线与P-R曲线十分类似，都是按照排序的顺序逐一按照正例预测，不同的是ROC曲线以TPR(True Positive Rate，真正例率)为横轴，纵轴为FPR(False Positive Rate，假正例率)，ROC偏重研究基于测试样本评估值的排序好坏。



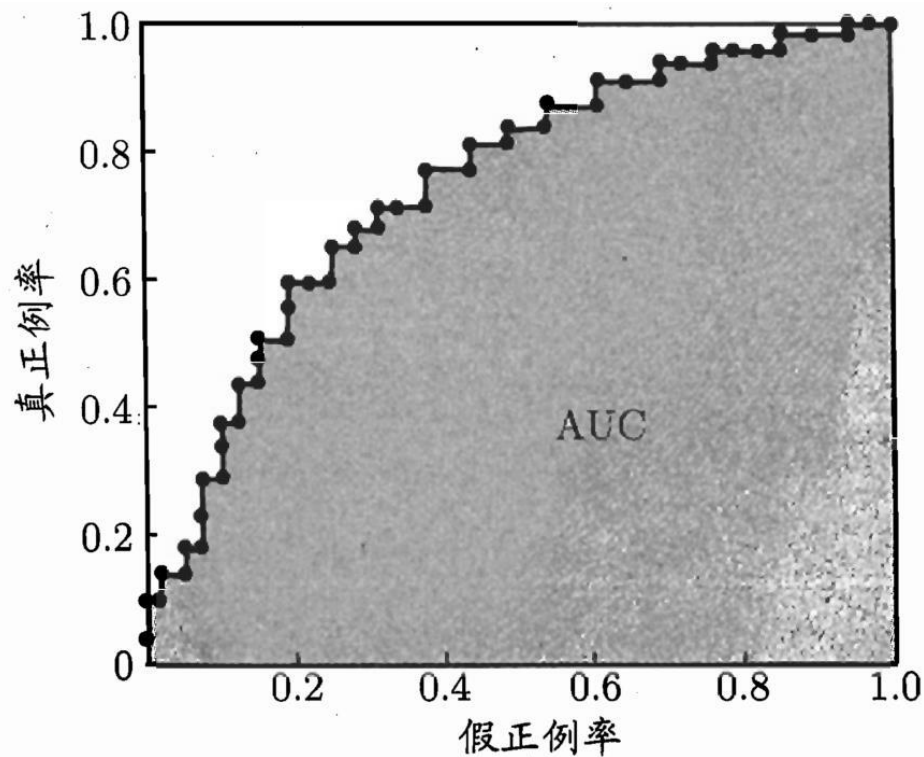
# ROC、AUC

$$TPR = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{TN + FP}$$



(a) ROC 曲线与 AUC



(b) 基于有限样例绘制的 ROC 曲线  
与 AUC