

第二讲 监督分类、模型评估与选择

授课老师：郭 迟 教授

guochi@whu.edu.cn

武汉大学测绘学院

2 0 2 1 . 1 1

- 1 监督学习的计算理论
 - 2 泛化与过拟合
 - 3 主要的模型评估方法
 - 4 性能度量
 - 5 比较检验
-

第二讲 监督分类、模型评估与选择



回顾机器学习基本概念

从机器学习的过程看，机器学习算法(或者机器学习模型)可分为：

- 有监督学习 (Supervised Learning)
- 无监督学习 (Unsupervised Learning)
- 半监督学习 (Semi-Supervised Learning)
- 强化学习 (Reinforcement Learning, RL)

机器学习算法以数据为对象，它通过提取数据特征，发现数据中的知识并抽象出数据模型，作出对数据的预测

机器学习算法能够有效的前提是同类数据（包括训练数据和测试数据等）具有**相同的统计规律性**这一基本假设

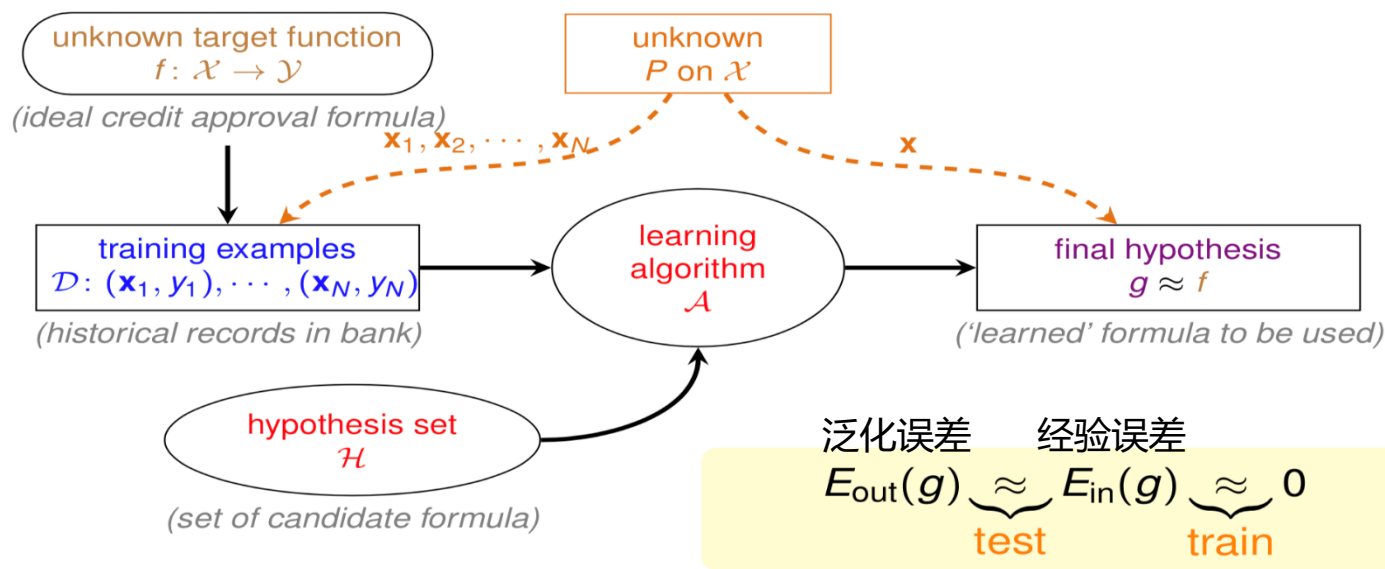
第二讲 监督分类、模型评估与选择



回顾机器学习基本概念

第一讲概念回顾：假设空间(H)

A 表示学习算法, f 表示学习的目标假设(可以是一个函数, 也可以是一个分布), H 表示假设空间, g 表示我们求解的用来预测的假设, g 是属于 H 的。 **机器学习的过程又可以理解为：通过算法 A , 在假设空间 H 中, 根据训练样本集 D , 选择最好的假设作为 g 使 g 近似于 f**



第一讲概念回顾：假设空间(H)

- $E_{in}(g)$ ，学到的假设 g 在训练样本(in-of-sample)上的损失，称为**经验误差**。
Training的过程希望经验误差 $E_{in}(g)$ 尽可能小
- $E_{out}(g)$ ，学到的假设 g 在除了训练样本外的其他所有样本(out-of-sample)上的损失，称为期望误差，也称**泛化误差**。在Test过程中希望 $E_{out}(g)$ 接近 $E_{in}(g)$

问题一：

f 作为目标假设其 $E_{out}(f) = 0$ 。为使 g 近似于 f ，即需要 $E_{out}(g) \approx E_{in}(g) \approx E_{out}(f) = 0$ 。但我们没法获得训练样本外的其他所有样本的，那么该如何计算 $E_{out}(g)$ 呢？

问题二：

给定一类假设，是否可以通过机器学习实现 $E_{out}(g) \approx E_{in}(g)$ （是否可学习？）

1. 霍夫丁Hoeffding不等式

$$\mathbf{P}[|E_{\text{out}}(g) - E_{\text{in}}(g)| > \varepsilon] \leq 2 \exp(-2 \varepsilon^2 N)$$



证明过程

- 当样本数量 N 越大时，样本期望与总体期望之差大于 ε 的概率上界趋近于0，即当样本数量 N 越来越多时，样本期望会接近总体期望，即他们的**概率近似正确** (probably approximately correct, PAC)
- 把经验误差当作样本期望，泛化误差当作总体期望，则对于ML来说，要提高预测的准确性，就需要增大样本



教材P16-17

2. 可学习条件

在我们的假设空间 H 中，往往有很多个假设(甚至无穷个)。那么对于假设空间 H 中的任意假设 h ， $E_{out}(h)$ 接近 $E_{in}(h)$ 之差大于 ε 的概率上界

$$\begin{aligned} & \mathbf{P}[|E(h_1)| > \varepsilon \cup |E(h_2)| > \varepsilon \cup |E(h_3)| > \varepsilon \cup \dots \cup |E(h_M)| > \varepsilon] \\ & \leq \mathbf{P}[|E(h_1)| > \varepsilon] + \mathbf{P}[|E(h_2)| > \varepsilon] + \dots + \mathbf{P}[|E(h_M)| > \varepsilon] \\ & \leq 2M \exp(-2 \varepsilon^2 N) \end{aligned}$$

在假设空间 H 中，对于任意一个假设 g ， $E_{out}(g)$ 与 $E_{in}(g)$ 之差大于 ε 的概率上界 $2M \exp(-2 \varepsilon^2 N)$ ，与训练样本数 N 和假设空间假设数 M 密切相关

2. 可学习条件

- 学习算法 A 能够从 H 选出的假设 g 满足 $E_{\text{out}}(g) \approx E_{\text{in}}(g) \approx 0$
 - 假设空间 H 中假设数 M 是有限的, 且训练样本数 N 足够大。即学习算法 A 从 H 选出的任意假设 g 都满足 $E_{\text{out}}(g) \approx E_{\text{in}}(g)$
- 当 M 较小时, 由于可选的 g 的数目少, 条件1不容易满足;
 - 当 M 较大时, 由于可选的 g 的数目 M 较大, 条件2不容易满足

在机器学习中, **假设数 M 在这两个核心条件中有着重要作用**。但是即便是“二维平面的直线划分正负类”这样的假设数都是无穷的, $2M \exp(-2 \varepsilon^2 N)$ 变为无穷大, 学习变得不可行, 这该怎么办呢?

第二讲 监督分类、模型评估与选择

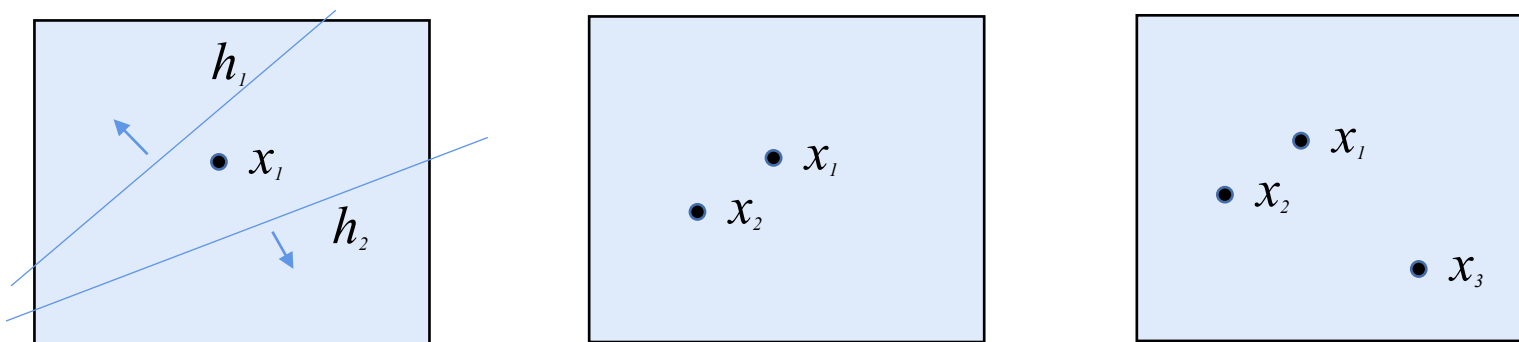


3. VC维 (Vapnik–Chervonenkis Dimension)



教材P16

- 有效假设数：在 M 个假设中，有很多假设都可以归为同一类



i.e. 二维空间的所有线性假设数(即直线条数)为 ∞ 。但如只存在1个数据点，则可以将这些假设分为两类，一类是把 x_1 判断为正例，另一类是把 x_1 判断为负例。提问：如果有2个数据点、3个数据点呢？

第二讲 监督分类、模型评估与选择



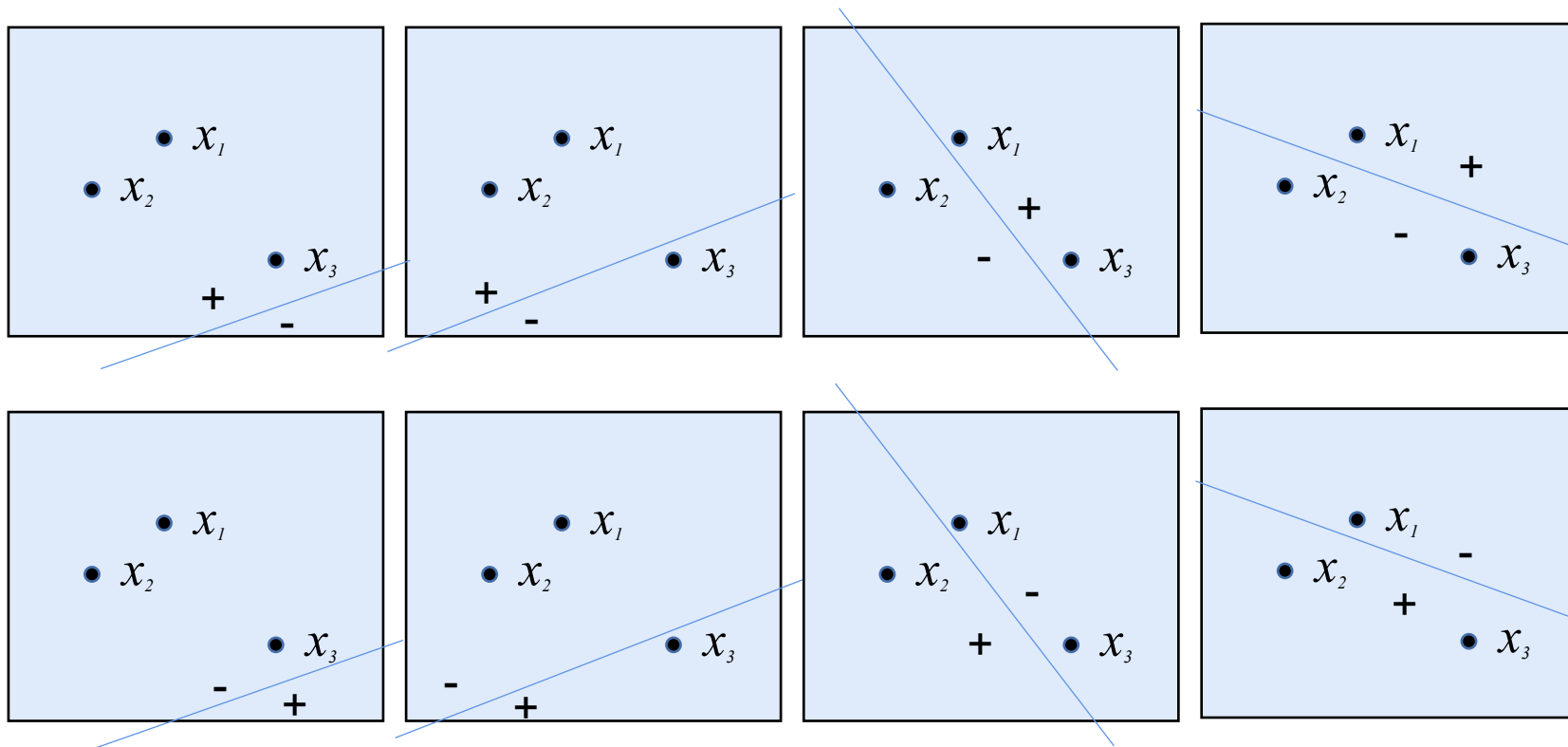
监督学习的计算理论

3. VC维 (Vapnik–Chervonenkis Dimension)



教材P16

➤ 有效假设数：在 M 个假设中，有很多假设都可以归为同一类



提问：如果有4个数据点呢？

第二讲 监督分类、模型评估与选择



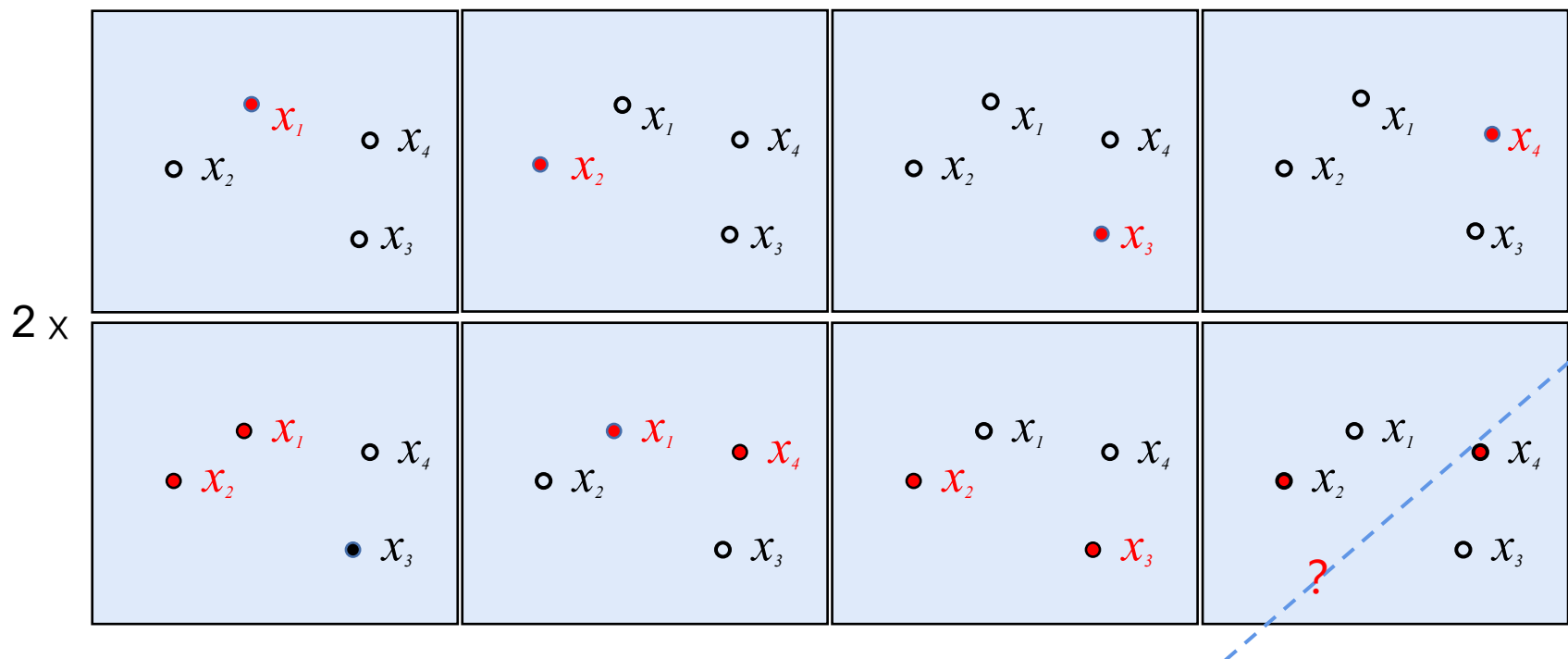
监督学习的计算理论

3. VC维 (Vapnik–Chervonenkis Dimension)



教材P16

➤ 有效假设数：在 M 个假设中，有很多假设都可以归为同一类



提问：如果有4个数据点呢？

3. VC维 (Vapnik–Chervonenkis Dimension)



教材P16

- 增长函数 $m_H(N)$ ：假设空间 H 对个任意 N 个样本所能赋予标记的最大可能数，其上界为 2^N ，每种标记成为一个对分 (dichotomy)
- H 散列 (shatter)： H 作用于大小为 N 的样本集 D 时，产生的对分数量等于 2^N 即 $m_H(N) = 2^N$ 时，就称 D 被 H 散列了

所以，二维平面的任意3个数据可以被直线散列 $m_H(3) = 2^3 = 8$ ，4个数据不能被直线散列 $m_H(4) = 14 < 2^4 = 16$

提问： 那么4个数据可以被什么样的假设散列呢？

第二讲 监督分类、模型评估与选择



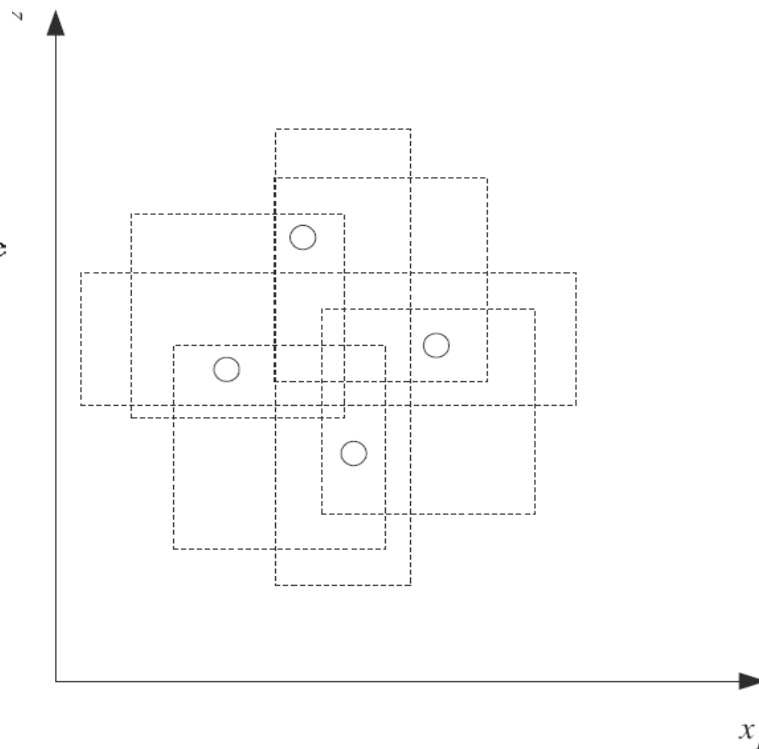
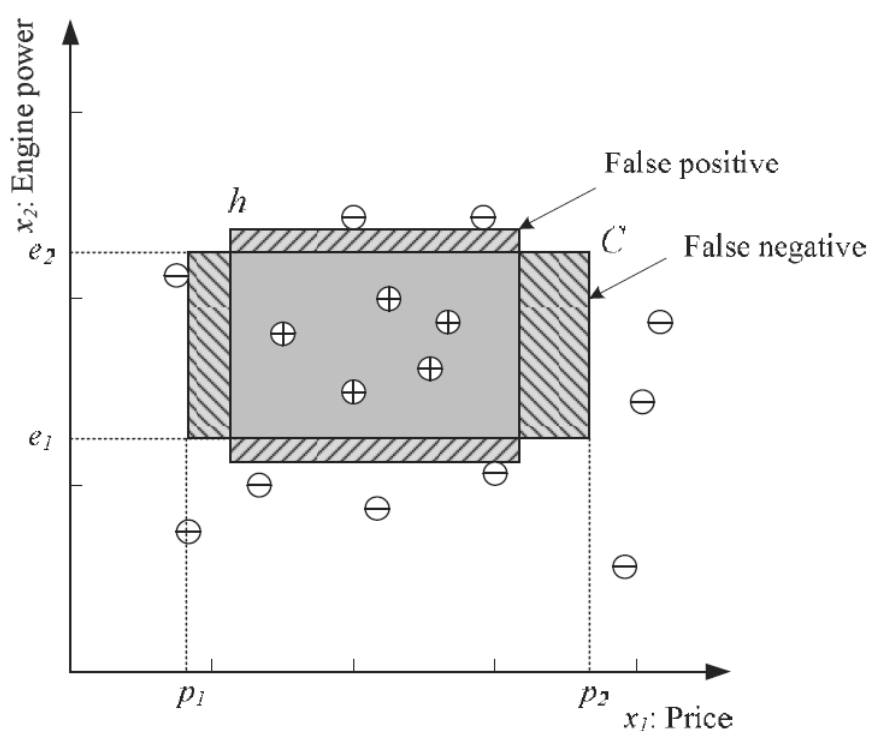
监督学习的计算理论

3. VC维 (Vapnik–Chervonenkis Dimension)



教材P16

- 轴平行的矩形能够散列二维空间的4个点



提问： 直线假设类、矩形假设类的参数分别是什么？

3. VC维 (Vapnik–Chervonenkis Dimension)



教材P16

- 中断点 (break point) : 对于假设空间 H 的增长函数 $m_H(N)$, 从 $N=1$ 逐渐增大到 k 时, 出现 $m_H(k) < 2^N$ 的情形, 则我们说 k 是该假设空间的中断点。对于任何大于中断点的数据集, H 都没有办法散列它
- 设中断点存在且为 k 的假设空间的增长函数 $m_H(N)$ 上界为 $B(N,k)$, 则 $B(N,k)$ 满足

$$m_H(N) \leq B(N,k) \leq \sum_{i=0}^{k-1} \binom{N}{i} \leq N^{k-1}$$



证明过程



《西瓜书》P276

则增长函数上界是一个最高幂次项为 N^{k-1} 的多项式。如果假设空间 H 存在中断点 k , 即 $m_H(N)$ 会被最高幂次为 $k-1$ 的多项式上界给约束住。那么, 当 N 足够大时, 对于 H 中的任意一个假设 g , $E_{in}(g)$ 都将接近于 $E_{out}(g)$, 即学习是可行的

3. VC维 (Vapnik–Chervonenkis Dimension)



教材P16

- VC维：可以被 H 散列的点的最大数据集大小称为 H 的VC维：

$$VC(H) = \max\{N: m_H(N) = 2^N\}$$

$VC(H) = k-1$ ，其中 k 是 H 的中断点。

$$\begin{aligned} & \mathbf{P}[|E(h_1)| > \varepsilon \cup |E(h_2)| > \varepsilon \cup |E(h_3)| > \varepsilon \cup \dots \cup |E(h_M)| > \varepsilon] \\ & \leq 2(2N)^{VC(H)} \exp(-\frac{1}{8} \varepsilon^2 N) \leq 4 m_H(2N) \exp(-\frac{1}{8} \varepsilon^2 N) \leq \end{aligned}$$



证明过程

重新审视可学习条件：

- 学习算法 A 能够从 H 选出的假设 g 满足 $E_{\text{out}}(g) \approx E_{\text{in}}(g) \approx 0$
- 假设空间 H 中 $VC(H)$ 是有限的，即学习算法 A 从 H 选出的任意假设 g 都满足 $E_{\text{out}}(g) \approx E_{\text{in}}(g)$

4. VC维对机器学习的理论指导意义

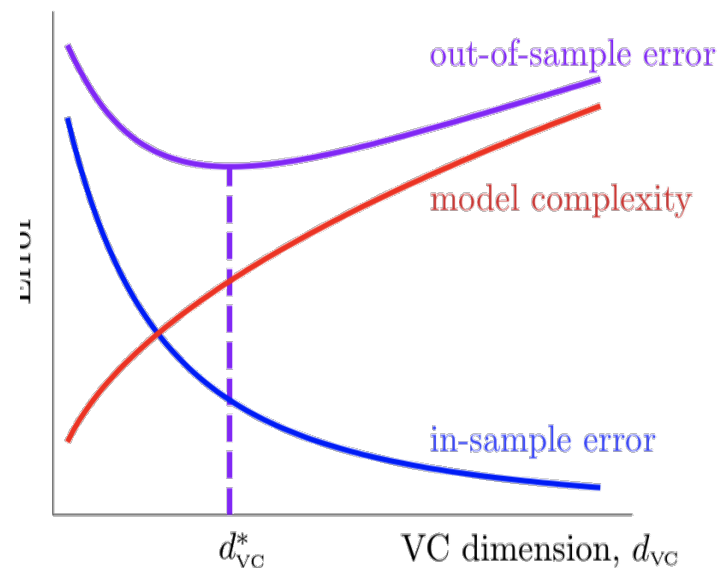


教材P17、P22

- **VC维用来度量假设类 H 的学习能力。**当固定样本数 N 时，随着VC维的上升， $E_{in}(g)$ 会不断降低，而模型复杂度 Ω 会不断上升，因此机器学习的模型选择要寻找一个合适的VC维使 $E_{out}(g)$ 最小

$$E_{in}(g) - \sqrt{\frac{8}{N} \ln\left(\frac{4(2N)^{VC(H)}}{\delta}\right)} \leq E_{out}(g) \leq E_{in}(g) + \sqrt{\frac{8}{N} \ln\left(\frac{4(2N)^{VC(H)}}{\delta}\right)}$$

- VC维越大，能学到的模型越复杂。VC维的大小与具体的学习方法 A 无关，与**数据集的分布无关**，与我们求解的目标函数 f 也无关，只与模型和假设空间有关
- 引用**三元权衡 (triple trade-off)** 理论总结



5. VC维对机器学习的工程指导意义

- 模型复杂时需要更多的训练数据。理论上数据规模 N 约为 $10000 \cdot VC(H)$ 这个量级，所以“VC维看起来比较悲观”
- 在实际工程中， N 取 $10 \cdot VC(H)$ 就能取得比较好效果。另外在训练中为了避免过拟合，一般都会加正则项。那加了正则项后的假设空间会受到一些限制，VC维也将变小（第9讲）
- 对于深度学习（神经网络），其VC维与神经元连接数有关。一个输入层1000、隐藏层1000，输出层为1的神经网络，VC维大约 $O(1000 \times 1000 \times 1)$ 。可见神经网络的VC维很高，因而它的表达能力非常强，可以用来处理任何复杂的分类问题。要充分训练该神经网络，所需样本量为10倍的VC维。这么大的训练数据量不容易达到，因此神经网络容易过拟合（第10讲）

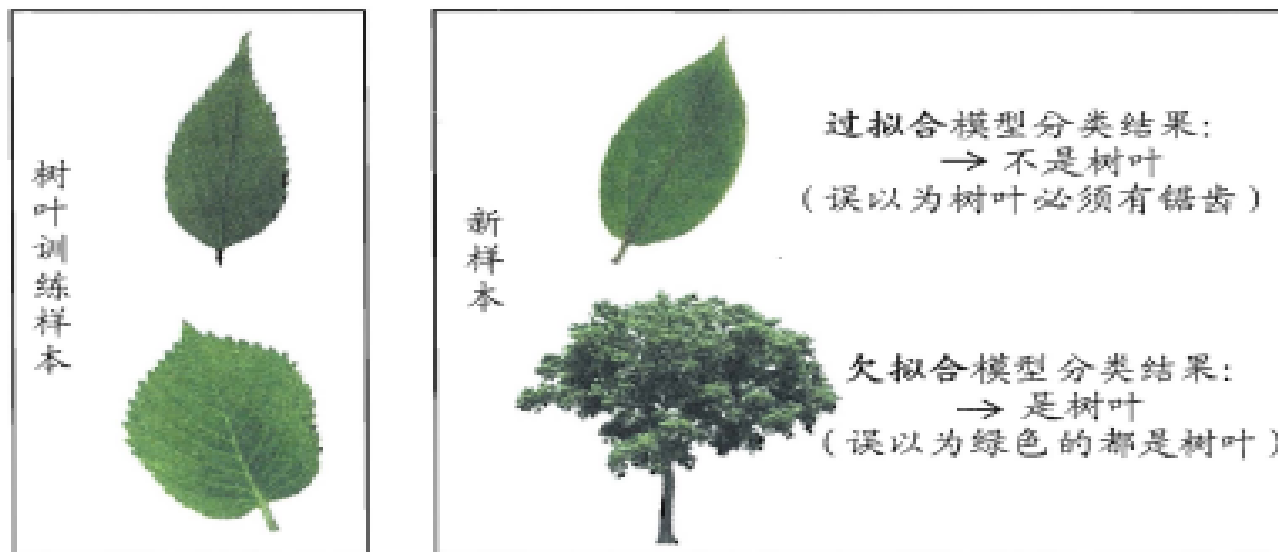
6. 有关VC维的思考题

- H 表示实数域的闭区间记作 $\{h_{[a,b]}\}$, 如果 $x \in [a,b]$ 则为正类, 否则为负类。问 H 的 VC 维是多少?
- H 表示二维平面的三角形假设类, 问 H 的 VC 维是多少?
- H 表示 \mathbb{R}^d 空间的线性超平面, 试着证明其 VC 维为 $d+1$

- 1 监督学习的计算理论
- 2 泛化与过拟合
- 3 主要的模型评估方法
- 4 性能度量
- 5 比较检验

1. 误差与过拟合(overfitting)

- 学习能力过强，以至于把训练样本的不太一般的特性都学到了，称为：过拟合。在过拟合问题中，经验误差小，但泛化（测试）误差大
- 学习能力太差，训练样本的一般性质尚未学好，称为：欠拟合 (underfitting)。在欠拟合问题中，训练误差和测试误差都比较大



过拟合、欠拟合的直观表示

1. 误差与过拟合(overfitting)

- **欠拟合**问题比较容易克服，例如增加迭代次数等
- **过拟合**问题还没有十分好的解决方案，是机器学习面临的关键障碍。各类学习算法都必然有针对过拟合的措施

基本指标

- **错误率 (error rate)**：分类错误的样本数占样本总数的比例；如果在 m 个样本中有 a 个样本分类错误，则错误率 $E\% = a/m$ ；
- **精度 (accuracy)**： $(1 - a/m)$ ，精度 = 1 - 错误率；
- **误差 (error)**：学习器对样本的实际预测结果与样本的真实值之间的差异。在训练集上的误差称为训练误差或经验误差；在测试集上的误差称为测试误差 (test error)；学习器在所有新样本上的误差称为泛化误差

第二讲 监督分类、模型评估与选择



泛化与过拟合

2. 泛化误差的构成



《西瓜书》P44

- 偏差-方差分解(bias-variance decomposition), “误差”包含了哪些因素? 从机器学习的角度看, “误差”从何而来? 对回归任务, 泛化误差可通过“偏差-方差及噪声”拆解为:

$$E(f; D) = \underbrace{bias^2(x)} + \underbrace{var(x)} + \underbrace{\varepsilon^2}$$

期望输出与真实输出的差别

$$bias^2(x) = (\bar{f}(x) - y)^2$$

同样大小的训练集的变动, 所导致的性能变化

$$var(x) = \mathbb{E}_D \left[(f(x; D) - \bar{f}(x))^2 \right]$$

表达了当前任务上任何学习算法所能达到的期望泛化误差下界

训练样本的标记与真实标记有区别

$$\varepsilon^2 = \mathbb{E}_D \left[(y_D - y)^2 \right]$$

泛化性能: 学习算法的能力、数据的充分性、学习任务本身的难度共同决定

2. 泛化误差的构成

- 一般而言，偏差与方差存在冲突：
- 训练不足时，学习器拟合能力不强，偏差主导
- 随着训练程度加深，学习器拟合能力逐渐增强，方差逐渐主导
- 训练充足后，学习器的拟合能力很强，方差主导
- 如果训练数据自身的、非全局的特性被学习器学到了，则发生过拟合

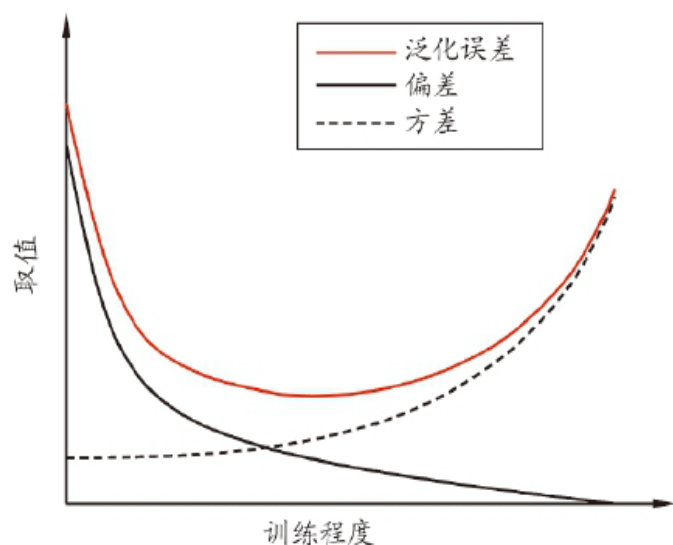
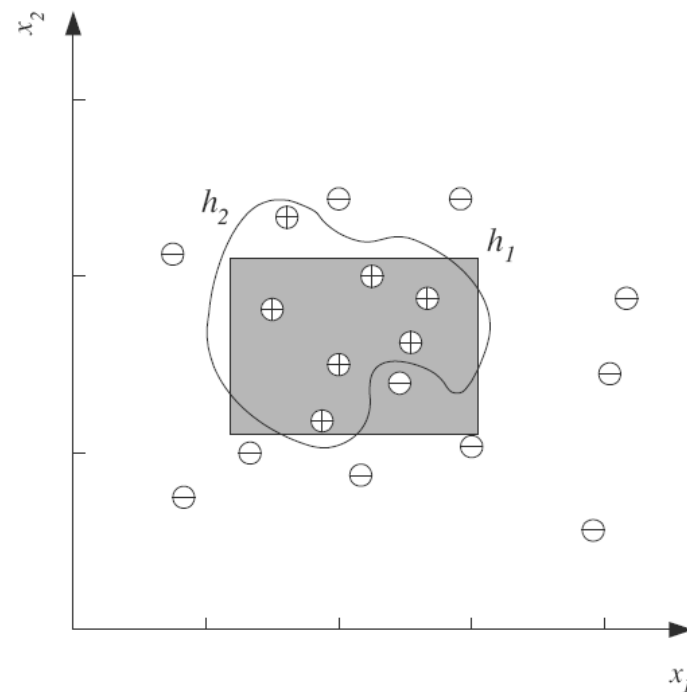


图 2.9 泛化误差与偏差、方差的关系示意图

3. 噪声

- 当有噪声时，正负实例之间没有简单的边界。
 - 利用复杂模型，更好地拟合数据，得到零误差。
 - 保持模型的简单性并允许一定误差的存在
- 噪声不可避免。其来源包括记录输入属性可能不准确；标记点可能有错。（指导噪声）；可能存在我们没有考虑到的附加属性。这些属性可能是隐藏的或潜在的，是不可以预测的；一种随机成分






- 1 监督学习的计算理论
- 2 泛化与过拟合
- 3 主要的模型评估方法
- 4 性能度量
- 5 比较检验

第二讲 监督分类、模型评估与选择



三个关键问题:

- 如何获得测试结果?  评估方法
- 如何评估性能优劣?  性能度量
- 如何判断实质差别?  比较检验

1. 模型选择

- 现实任务中，往往有多种学习算法可供选择，甚至同一学习算法，当使用不同参数配置时，会产生不同的模型。如何选择学习算法，使用哪种参数配置是机器学习应用的首要面对问题
- 理想的解决方案：对候选模型的泛化误差进行评估，然后选择泛化误差最小的模型。但是，泛化误差无法直接获得，而训练误差有因为过拟合现象的存在而不适合作为标准，现实中的模型到底如何评估与选择呢？

关键：怎么获得“测试集” (test set) ？

测试集应该与训练集“互斥”

1. 模型选择



教材P320-P321

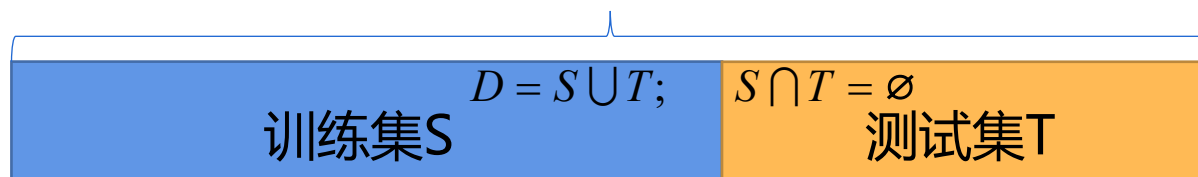
假设只有一个包含 m 个样例的数据集 $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_m, y_m)\}$,

通过对 D 进行适当的处理, 从中产生出训练集 S 和测试集 T , 通常采用以下:

- 留出法 (hold-out)
- 交叉验证法 (cross validation)
- 自助法 (bootstrap)

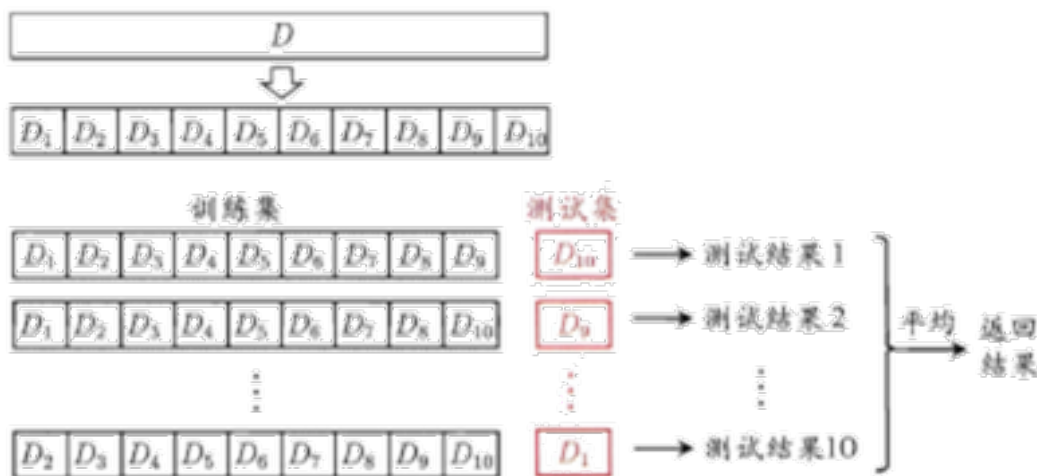
2. 留出法

拥有的数据集D



- 保持数据分布一致性（例如：分层采样）；若 S 、 T 中样本类别比例差别很大，则误差估计将由于训练/测试数据分布的差异而产生偏差；
- 多次重复划分（例如：100次随机划分），重复进行实验评估后取平均值作为留出法的评估结果；
- 测试集不能太大、不能太小（例如：1/5~1/3）
 - 若 S 太大， T 太小—>评估结果不够稳定准确；
 - 若 T 多，则 S 与 D 差别太大，被评估模型结果与用 D 训练的模型差异太大，降低了评估结果的保真性（fidelity）

2. 交叉验证法



- 将数据集 D 划分为 k 个大小相同的互斥子集，满足 $D = D_1 \cup D_2 \cup \dots \cup D_k$ ， $D_i \cap D_j = \emptyset$ ($i \neq j$)，保持数据分布的一致性，采用分层抽样的方法获得这些子集。每次用 $k-1$ 个子集的并集作为训练集，余下的那个子集作为测试集。这样就有 k 种训练集/测试集划分的情况，从而可进行 k 次训练和测试，最终返回 k 次测试结果的均值。

2. 交叉验证法

假定数据集 D 中包含 m 个样本，若令 $k=m$ ，则得到了交叉验证法的一个特例：

留一法 (Leave-One-Out, LOD)

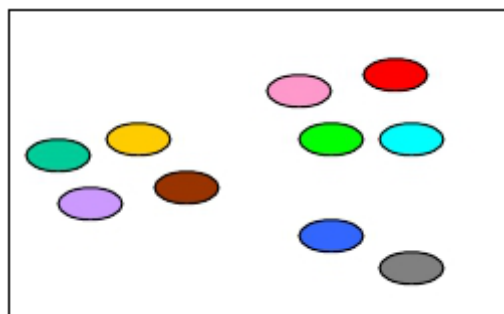
➤ 优点：准确

留一法不受随机样本划分的方式影响，因为 m 个样本只有唯一的方式划分为 m 个子集——每个子集包含一个样本；留一法使用的训练集与初始数据集相比只少了一个样本，这就使得在绝大多数情况下，留一法中被实际评估的模型与期望评估的用 D 训练出的模型很相似；

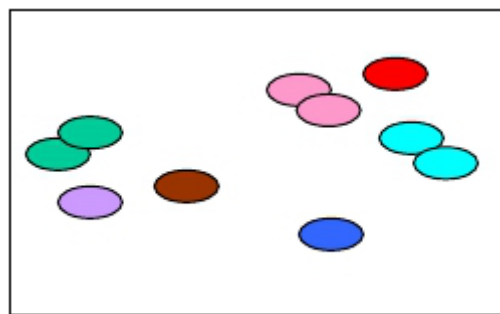
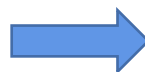
➤ 缺点：开销大

在数据集较大时，训练 m 个模型的计算开销可能是难以忍受的；

3. 自助法



训练集与原样本集同规模



数据分布有所改变

- “自助采样” (bootstrap sampling), 亦称 “有放回采样”、“可重复采样”。基本思想是：给定包含 m 个样本的数据集 D ，每次随机从 D 中挑选一个样本，将其拷贝放入 D' ，然后再将该样本放回初始数据集 D 中，使得该样本在下次采样时仍有可能被采到。重复执行 m 次，就可以得到了包含 m 个样本的数据集 D' 。

*在 m 次采样中，样本始终不被采到的概率极限 $\lim_{m \rightarrow \infty} \left(1 - \frac{1}{m}\right)^m \mapsto \frac{1}{e} \approx 0.368$

3. 自助法

D' 作为训练集, $D \setminus D'$ 用作测试集。实际评估模型与期望评估模型都使用了 m 个训练样本, 但是仍有 $1/3$ 没在训练集中出现的样本用于测试。这样的测试结果, 被称为“包外估计” (out-of-bag estimate)

➤ 优点:

自助法适用于数据集较小、难以有效划分训练/测试集时很有效。此外, 自助法能从初始数据集中产生多个不同的训练集, 有利于集成学习方法

➤ 缺点:

自助法改变了初始数据集的分布, 会引入估计偏差

4. 参数调节 (parameter tuning)

- 大多数学习算法都有些参数(parameter) 需要设定, 参数配置不同, 学习所得模型的性能往往有显著差别, 这就是通常所说的“ 参数调节” 或简称“ 调参” 。
- 机器学习常用参数: (均需要产生多个模型后, 基于某种评估方法来择优)
 - 算法参数 (超参数), 数目10以内, 通常人工设定;
 - 模型参数, 数目众多, 通过学习确定, 如深度学习模型;
- 通常把学得模型在实际使用中遇到的数据称为测试数据, 为了加以区分, 模型评估与选择中用于评估测试的数据集常称为 “验证集 (Validation set) ” 。把训练数据另外划分为训练集和验证集, 基于验证集上的性能来进行模型的选择和调参。

区别: 训练集 vs. 测试集 vs. 验证集

4. 参数调节 (parameter tuning)




- 学习算法的很多参数是在实数范围内取值，因此，对每种参数取值都训练出模型来是不可行的。常用的做法是：对每个参数选定一个范围和步长 λ ，这样使得学习的过程变得可行。例如：假定算法有3个参数，每个参数仅考虑5个候选值，这样对每一组训练/测试集就有 $5 \times 5 \times 5 = 125$ 个模型需考察。很多强大的学习算法有不少参数要设定，这将使得调参工作量巨大，所以，在很多情况下，参数调得好不好往往对最终模型性能有关键影响

- 1 监督学习的计算理论
- 2 泛化与过拟合
- 3 主要的模型评估方法
- 4 性能度量
- 5 比较检验

第二讲 监督分类、模型评估与选择



三个关键问题:

- 如何获得测试结果?  评估方法
- 如何评估性能优劣?  性能度量
- 如何判断实质差别?  比较检验

1. 均方误差

性能度量是衡量模型泛化能力的评价标准，反映了任务需求，在对比不同模型能力是，使用不同的性能度量会导致不同的评判结果

最常见的性能度量 **Loss**

“均方误差” (mean squared error)



教材P24

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

更一般的, 对于数据分布 \mathcal{D} 和概率密度函数 $p(\cdot)$, 均方误差可描述为

$$E(f; \mathcal{D}) = \int_{\mathbf{x} \sim \mathcal{D}} (f(\mathbf{x}) - y)^2 p(\mathbf{x}) d\mathbf{x} .$$

2. 错误率与精度

在分类任务中，即预测离散值的问题，最常用的是错误率和精度

错误率与精度

错误率定义为

$$E(f; D) = \frac{1}{m} \sum_{i=1}^m \mathbb{I}(f(\mathbf{x}_i) \neq y_i) .$$

精度则定义为

$$\begin{aligned} \text{acc}(f; D) &= \frac{1}{m} \sum_{i=1}^m \mathbb{I}(f(\mathbf{x}_i) = y_i) \\ &= 1 - E(f; D) . \end{aligned}$$

第二讲 监督分类、模型评估与选择



性能度量

3. 查准率/查全率



教材P322-P324

在分类任务中，针对分类结果混淆矩阵，最常用的是查准率/查全率

查准率/查全率

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)

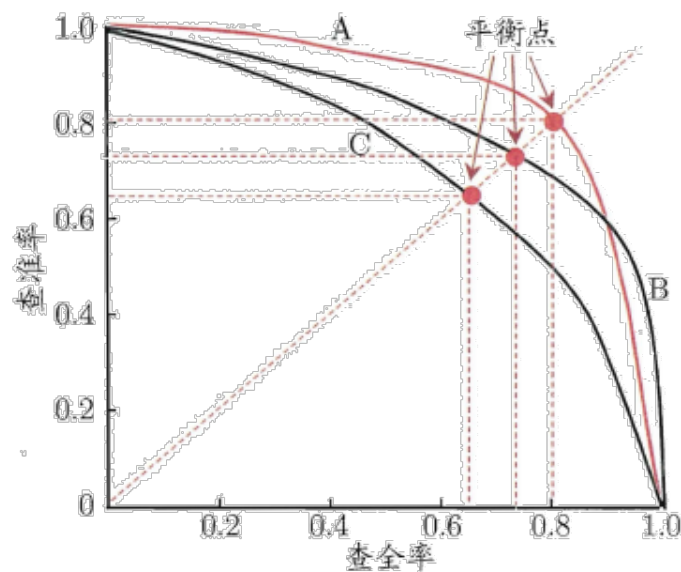
查准率 P 与查全率 R 分别定义为

$$P = \frac{TP}{TP + FP},$$

$$R = \frac{TP}{TP + FN}.$$

3. 查准率/查全率

- “P-R曲线”：根据学习器的预测结果（一般为一个实值或概率）对测试样本进行排序，将最可能是“正例”的样本排前，最不可能是“正例”的排后，按此顺序逐个把样本作为“正例”进行预测，每次计算出当前的P值和R值



- 若一个学习器A的P-R曲线被另一个学习器B的P-R曲线完全包住，则称：B的性能优于A。若A和B的曲线发生了交叉，则谁的曲线下的面积大，谁的性能更优。
- “平衡点”（Break-Event Point，简称BEP），即当 $P=R$ 时的取值，平衡点的取值越高，性能更优

3. 查准率/查全率

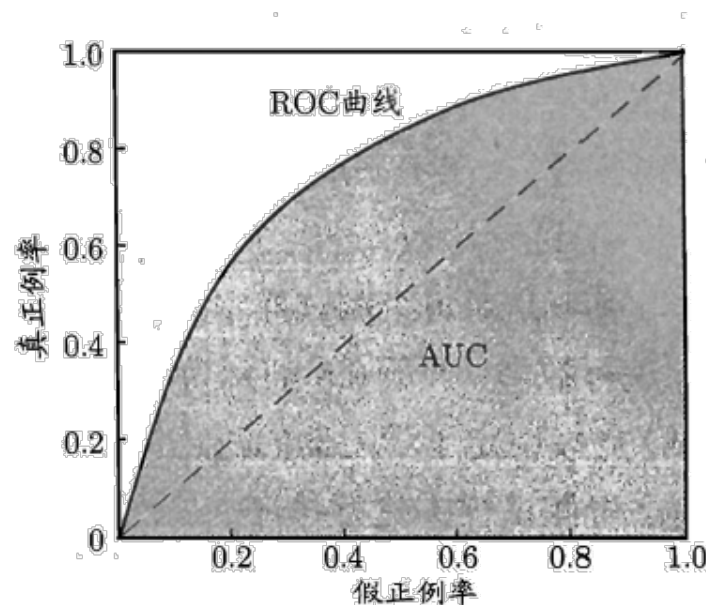
- ROC (Receiver Operating Characteristic) 曲线: 接受者操作特性曲线。学习器对测试样本的评估结果一般为一个实值或概率, 设定一个阈值, 大于阈值为正例, 小于阈值为负例。因此这个阈值决定了学习器的泛化性能。ROC曲线其与坐标轴围成的面积AUC越大性能越优

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, 纵轴: 是“真正率” (True Positive Rate, 简称TPR) ;

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

横轴为“假正率” (False Positive Rate, 简称FPR) ;

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



(a) ROC 曲线与 AUC

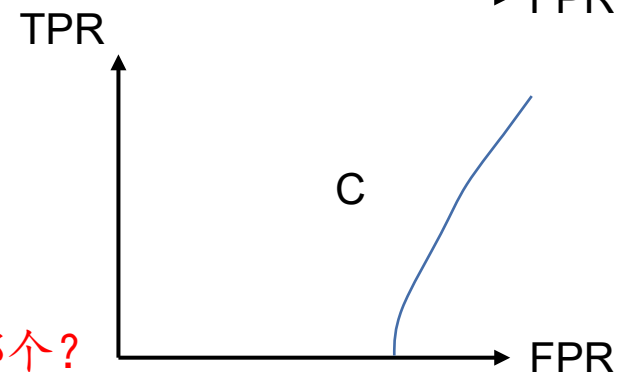
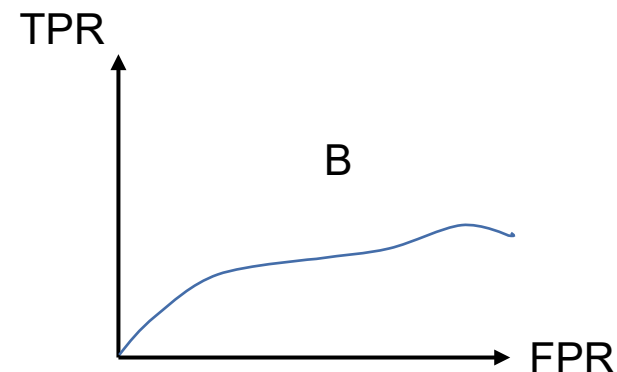
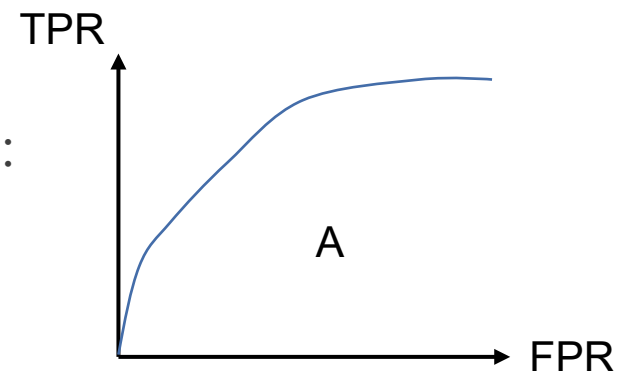
第二讲 监督分类、模型评估与选择



性能度量

3. 查准率/查全率




➤ ROC (Receiver Operating Characteristic) 曲线:



提问：你们认为我校门禁人脸识别的ROC曲线更像哪个？

- 1 监督学习的计算理论
 - 2 泛化与过拟合
 - 3 主要的模型评估方法
 - 4 性能度量
 - 5 比较检验
-

三个关键问题:

- 如何获得测试结果?  评估方法
- 如何评估性能优劣?  性能度量
- 如何判断实质差别?  比较检验

1. 假设检验 (hypothesis test)

- 统计假设检验为学习器性能比较提供了量化方法，可以度量学习器泛化错误率的分布是否符合某种判断或猜想
- 在比较学习器泛化性能的过程中，统计假设检验为学习器性能比较提供了重要依据，即若A在某测试集上的性能优于B，那A学习器比B好的把握有多大

假设检验：

在总体的分布函数完全未知或只知道其形式但不知道其参数的情况下，为了推断总体的某些性质，首先提出关于总体的假设，然后根据样本所提供的信息对该假设做出“是”或“否”的结论性判断

2. 单个算法上的假设检验

- 单验证集检验
 - 二项检验
 - 单总体 t 检验：检验一个样本平均数与一个已知的总体平均数的差异是否显著
- 多验证集检验
 - 双总体 t 检验：检验两个样本平均数与其各自所代表的总体的差异是否显著

3. 比较两个分类算法

➤ 两学习器比较

□ 交叉验证 t 检验 (基于成对 t 检验)

k 折交叉验证; 5x2交叉验证

□ McNemar 检验 (基于列联表, 卡方 χ^2 检验)

➤ 多学习器比较

□ Friedman + Nemenyi

- Friedman检验 (基于序值, F检验; 判断“是否都相同”)
- Nemenyi 后续检验 (基于序值, 进一步判断两两差别)