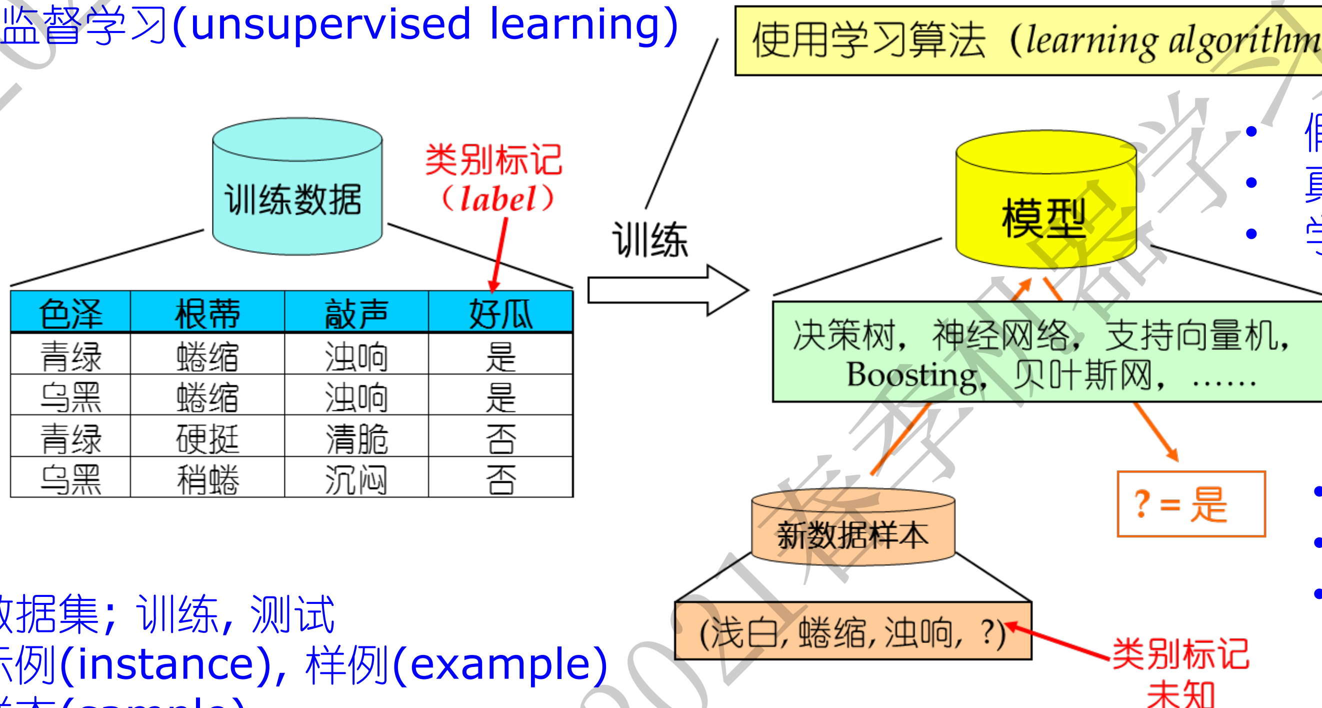
一、绪论

主讲教师：周志华

基本术语

- 监督学习(supervised learning)
- 无监督学习(unsupervised learning)



色泽	根蒂	敲声	好瓜
青绿	蜷缩	浊响	是
乌黑	蜷缩	浊响	是
青绿	硬挺	清脆	否
乌黑	稍蜷	沉闷	否

- 假设(hypothesis)
- 真相(ground-truth)
- 学习器(learner)

- 分类, 回归
- 二分类, 多分类
- 正类, 反类

- 数据集; 训练, 测试
- 示例(instance), 样例(example)
- 样本(sample)
- 属性(attribute), 特征(feature); 属性值
- 属性空间, 样本空间, 输入空间
- 特征向量(feature vector)
- 标记空间, 输出空间

- 未见样本(unseen instance)
- 未知 “分布”
- 独立同分布(i.i.d.)
- **泛化(generalization)**

假设空间

表 1.1 西瓜数据集

编号	色泽	根蒂	敲声	好瓜
1	青绿	蜷缩	浊响	是
2	乌黑	蜷缩	浊响	是
3	青绿	硬挺	清脆	否
4	乌黑	稍蜷	沉闷	否

$\boxed{(\text{色泽}=?) \land (\text{根蒂}=?) \land (\text{敲声}=?) \leftrightarrow \text{好瓜}}$

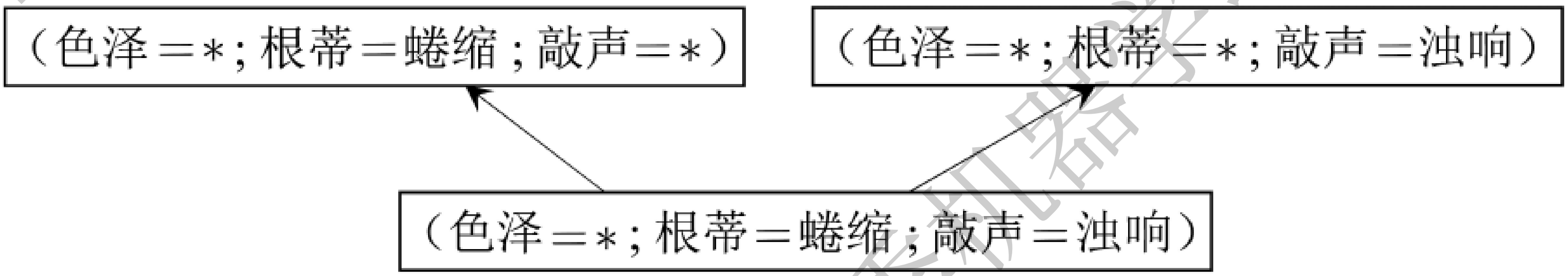
学习过程 → 在所有假设(hypothesis)组成的空间中进行**搜索**的过程

目标：找到与训练集“匹配”(fit)的假设

假设空间的大小：(n1+1) x (n2+1) x (n3+1) + 1

版本空间

版本空间(version space)：与训练集一致的假设集合



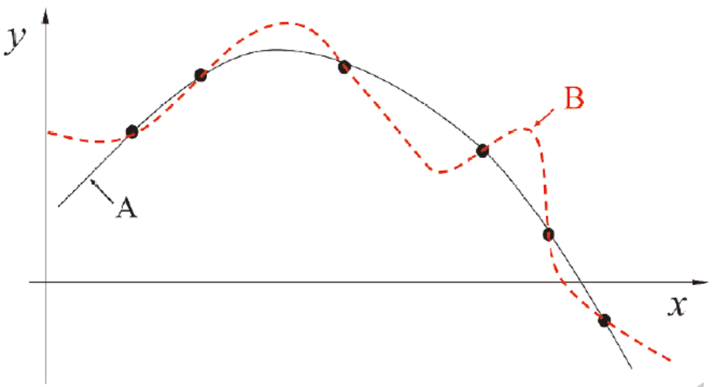
在面临新样本时，会产生不同的输出

例如：(青绿; 蜷缩; 沉闷)

应该采用哪一个模型(假设)?

归纳偏好 (inductive bias)

机器学习算法在学习过程中对某种类型假设的偏好



A更好？
B更好？

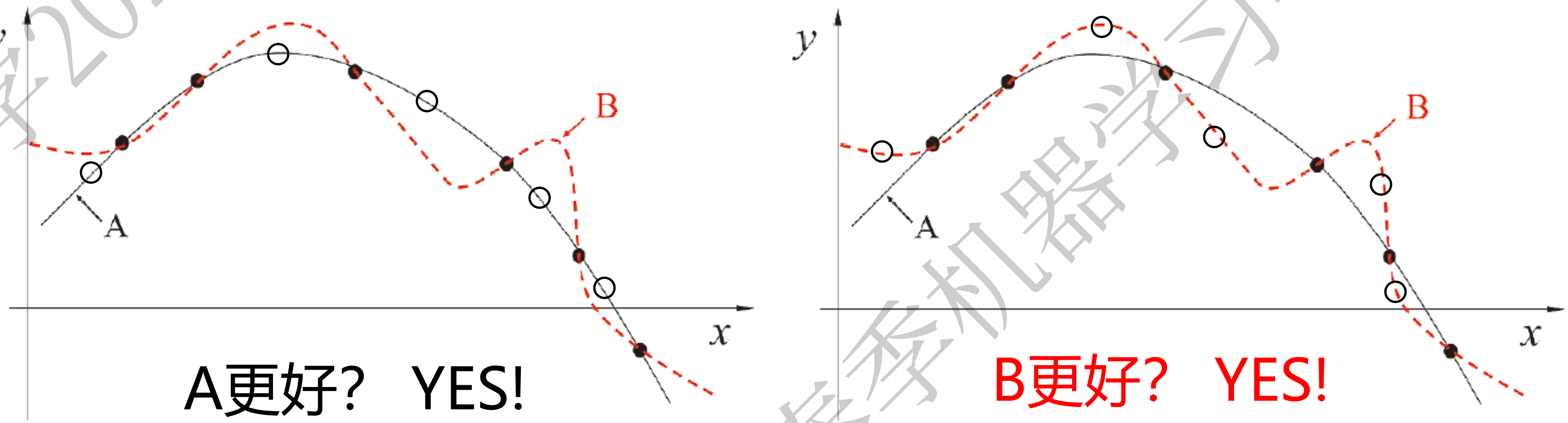
一般原则：
奥卡姆剃刀
(Ocam's razor)

任何一个有效的机器学习算法必有其偏好

学习算法的归纳偏好是否与问题本身匹配，
大多数时候直接决定了算法能否取得好的性能！

哪个算法更好？

黑点：训练样本；白点：测试样本



没有免费的午餐！

NFL定理：一个算法 $\mathfrak{L}_a$ 若在某些问题上比另一个算法 $\mathfrak{L}_b$ 好，必存在另一些问题，$\mathfrak{L}_b$ 比 $\mathfrak{L}_a$ 好

NFL定理

简单起见，假设样本空间 $\mathcal{X}$ 和假设空间$\mathcal{H}$ 离散, 令 $P(h|X, \mathfrak{L}_a)$ 代表算法 $\mathfrak{L}_a$ 基于训练数据 $\boldsymbol{X}$ 产生假设 h 的概率，f 代表要学的目标函数，$\mathfrak{L}_a$ 在训练集之外所有样本上的总误差为

$$E_{ote}(\mathfrak{L}_a|X, f) = \sum_h \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x})\,\mathbb{I}\left(h\left(\boldsymbol{x}\right) \neq f\left(\boldsymbol{x}\right)\right) P\left(h \mid X, \mathfrak{L}_a\right)$$

考虑二分类问题，目标函数可以为任何函数 $\mathcal{X} \mapsto \{0,1\}$，函数空间为 $\{0,1\}^{|\mathcal{X}|}$，对所有可能的 f 按均匀分布对误差求和，有

$$\sum_f E_{ote}(\mathfrak{L}_a|X, f) = \sum_f \sum_h \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x})\,\mathbb{I}(h(\boldsymbol{x}) \neq f(\boldsymbol{x}))\, P(h \mid X, \mathfrak{L}_a)$$

NFL定理

考虑二分类问题，目标函数可以为任何函数 $\mathcal{X} \mapsto \{0,1\}$，函数空间为 $\{0,1\}^{|\mathcal{X}|}$，对所有可能的 f 按均匀分布对误差求和，有

$$
\begin{aligned}
\sum_f E_{ote}(\mathfrak{L}_a|X,f) &= \sum_f \sum_h \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x})\, \mathbb{I}(h(\boldsymbol{x}) \neq f(\boldsymbol{x}))\, P(h \mid X, \mathfrak{L}_a) \\
&= \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x}) \sum_h P(h \mid X, \mathfrak{L}_a) \sum_f \mathbb{I}(h(\boldsymbol{x}) \neq f(\boldsymbol{x})) \\
&= \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x}) \sum_h P(h \mid X, \mathfrak{L}_a) \frac{1}{2} 2^{|\mathcal{X}|} \\
&= \frac{1}{2} 2^{|\mathcal{X}|} \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x}) \sum_h P(h \mid X, \mathfrak{L}_a) \\
&= 2^{|\mathcal{X}|-1} \sum_{\boldsymbol{x}\in\mathcal{X}-X} P(\boldsymbol{x}) \cdot 1
\end{aligned}
$$

总误差与学习算法无关！ $\Rightarrow$ **所有算法同样好！**

NFL定理的寓意

NFL定理的重要前提：

所有“问题”出现的机会相同、或所有问题同等重要

实际情形并非如此；我们通常只关注自己正在试图解决的问题

脱离具体问题，空泛地谈论“什么学习算法更好”毫无意义！

具体问题，具体分析！

现实机器学习应用中 ……

把机器学习的 “十大算法” “二十大算法” 都弄熟，

逐个试一遍，是否就 “止于至善” 了 ?

NO !

机器学习并非 “十大套路” “二十大招数” 的简单堆积

现实任务千变万化，

以有限的 “套路” 应对无限的 “问题” ，焉有不败 ?

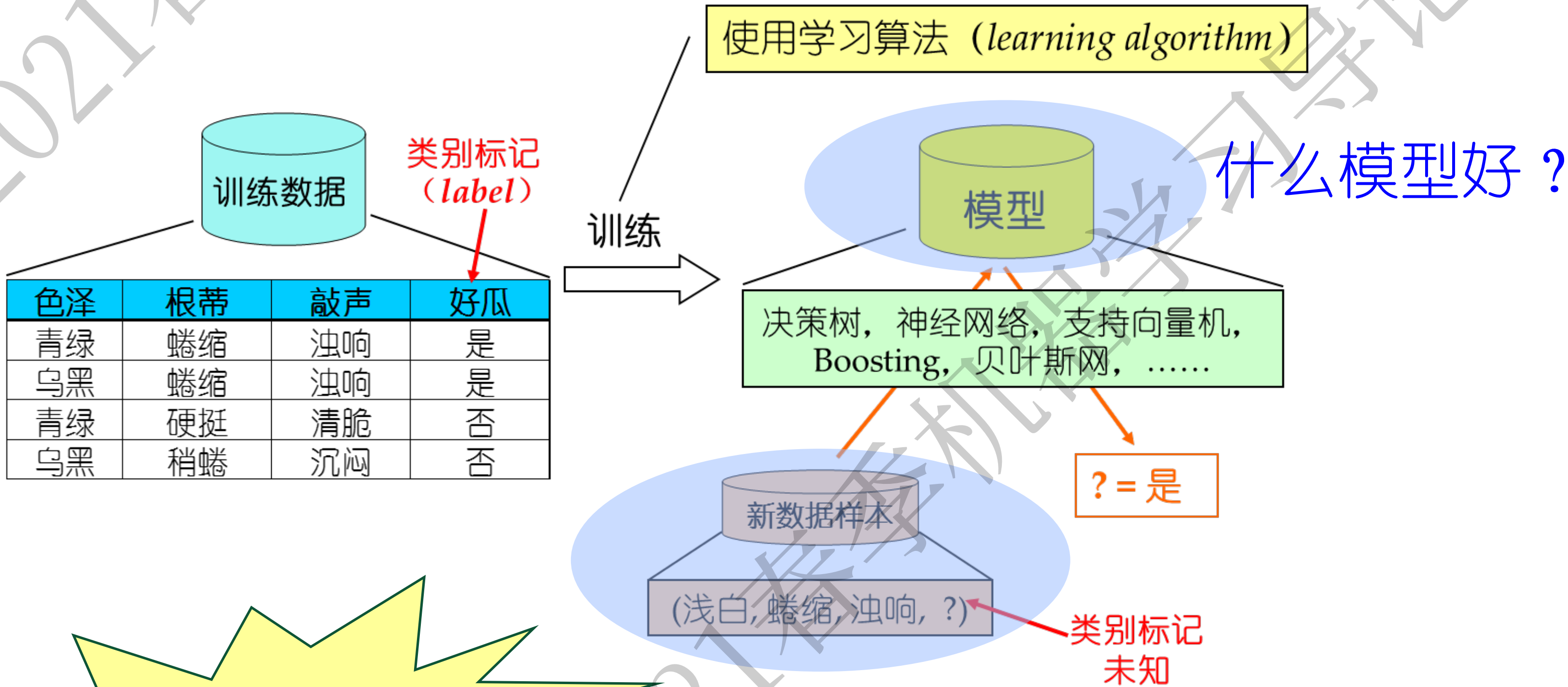
最优方案往往来自：**按需设计、度身定制**

前往第二站……

二、模型评估与选择

主讲教师：周志华

典型的机器学习过程



色泽	根蒂	敲声	好瓜
青绿	蜷缩	浊响	是
乌黑	蜷缩	浊响	是
青绿	硬挺	清脆	否
乌黑	稍蜷	沉闷	否

能很好地适用于 unseen instance

例如，**错误率**低、**精度**高

然而，我们手上没有 unseen instance，……

泛化误差 vs. 经验误差

泛化误差：在“未来”样本上的误差

经验误差：在训练集上的误差，亦称“训练误差”

- 泛化误差越小越好
- 经验误差是否越小越好？

NO! 因为会出现“过拟合”(overfitting)

过拟合 (overfitting) vs. 欠拟合 (underfitting)

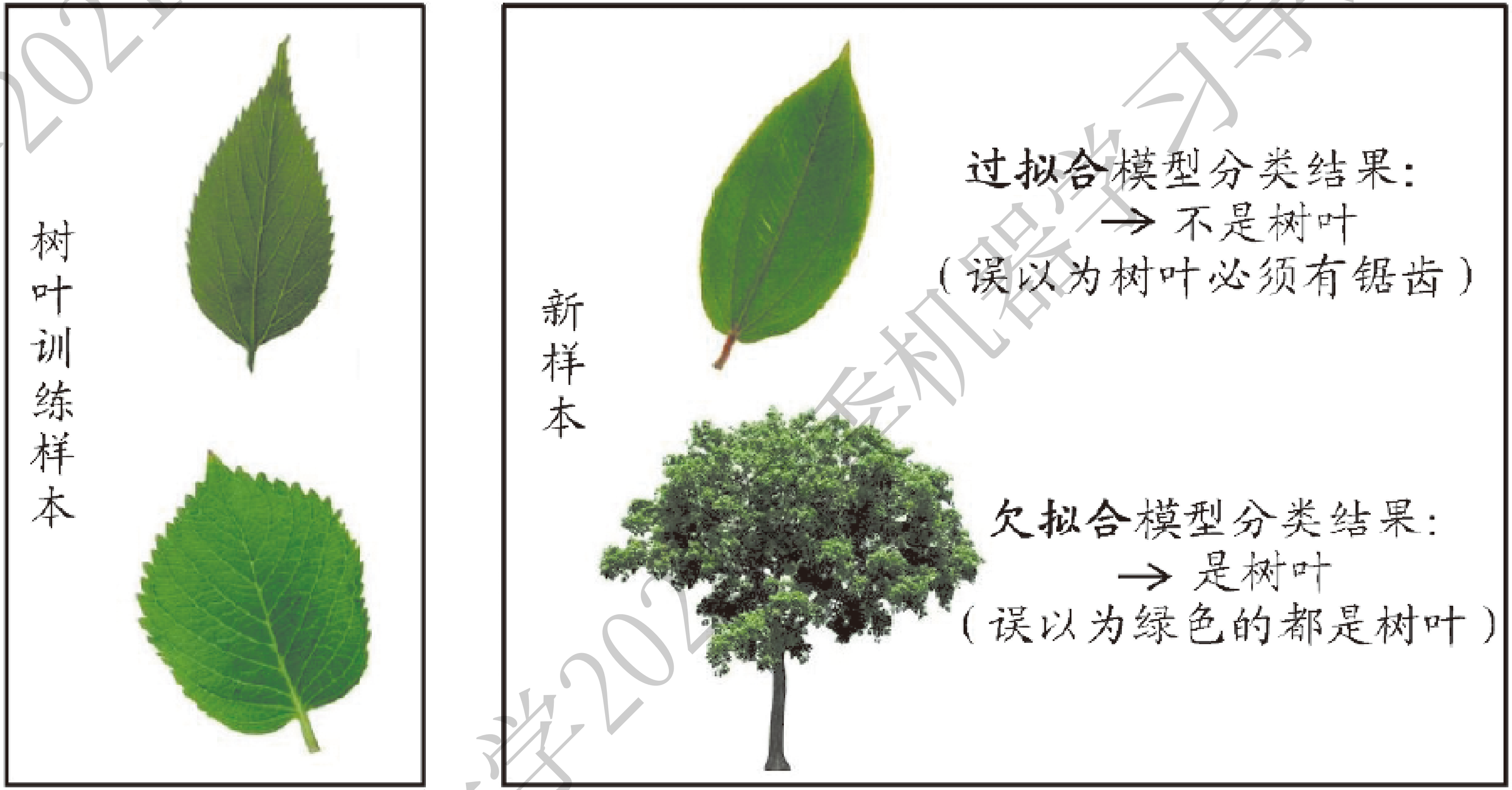


图 2.1 过拟合、欠拟合的直观类比

模型选择 (model selection)

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

评估方法

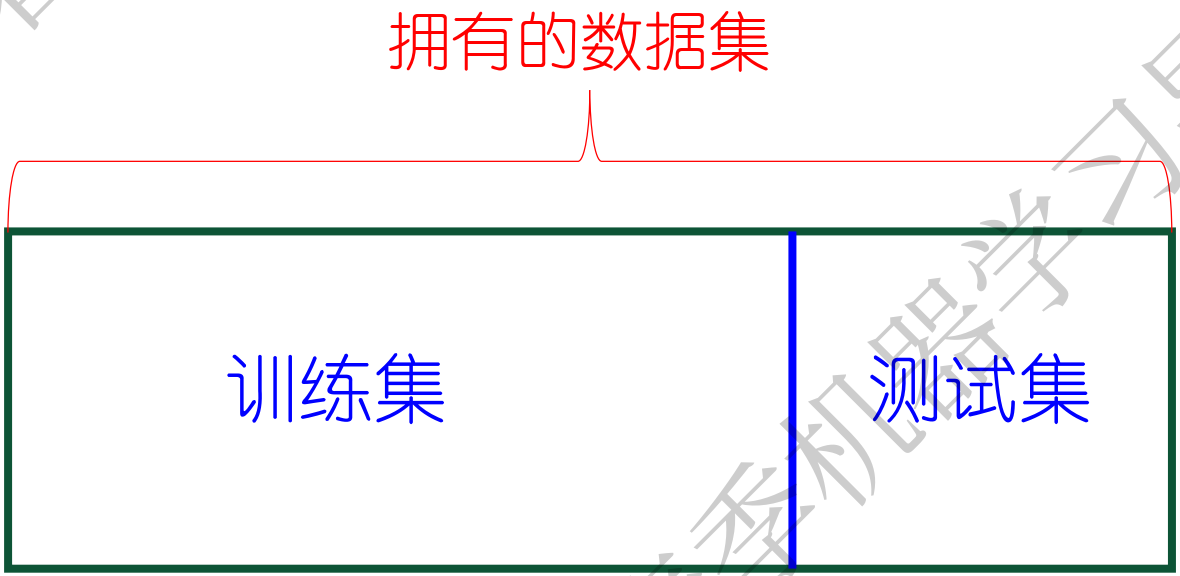
关键：怎么获得“测试集”(test set)？

测试集应该与训练集“互斥”

常见方法：

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

留出法



注意：

- 保持数据分布一致性（例如：分层采样）
- 多次重复划分（例如：100次随机划分）
- 测试集不能太大、不能太小（例如：1/5~1/3）

k-折交叉验证法

若 k = m，则得到“留一法”
(leave-one-out, LOO)

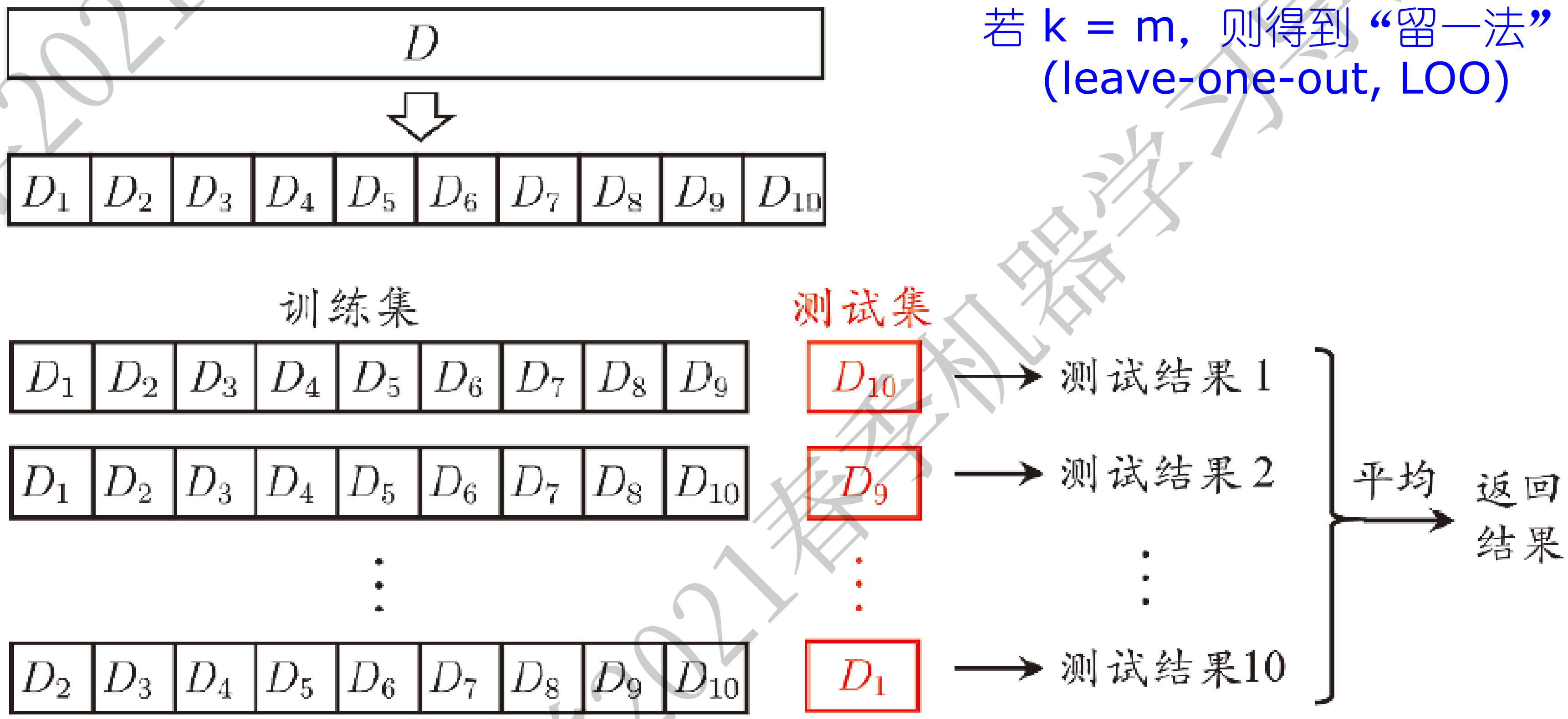
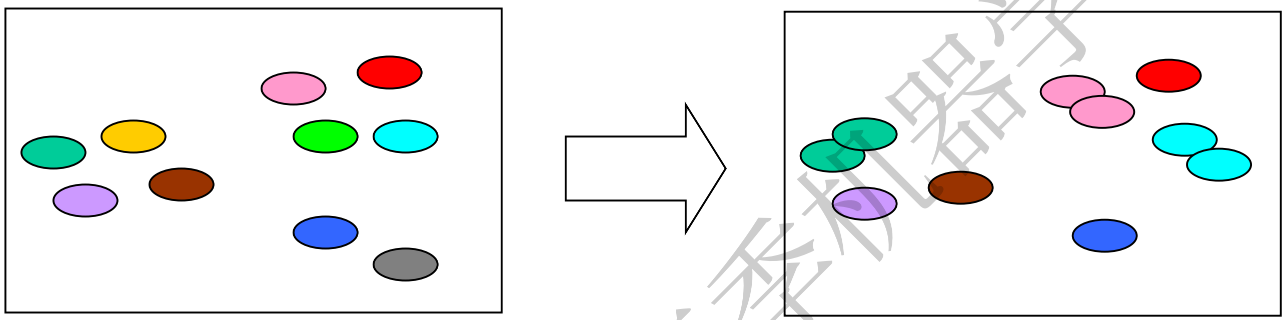


图 2.2 10 折交叉验证示意图

自助法

基于“自助采样” (bootstrap sampling)

亦称“有放回采样”、“可重复采样”



约有 36.8% 的样本不出现

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

“包外估计” (out-of-bag estimation)

- 训练集与原样本集同规模
- 数据分布有所改变

“调参”与最终模型

算法的参数：一般由人工设定，亦称“超参数”

模型的参数：一般由学习确定

调参过程相似：先产生若干模型，然后基于某种评估方法进行选择

参数调得好不好对性能往往对最终性能有关键影响

区别：训练集 vs. 测试集 vs. 验证集 (validation set)

算法参数选定后，要用“训练集+验证集”重新训练最终模型

模型选择 (model selection)

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

性能度量

性能度量(performance measure)是衡量模型泛化能力的评价标准，反映了任务需求

使用不同的性能度量往往会导致不同的评判结果

什么样的模型是“好”的，不仅取决于算法和数据，还取决于任务需求

- 回归(regression) 任务常用均方误差：

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

错误率 vs. 精度

- 错误率：

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

- 精度：

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

查准率 vs. 查全率

表 2.1 分类结果混淆矩阵

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： $$P = \frac{TP}{TP + FP}$$
- 查全率： $$R = \frac{TP}{TP + FN}$$

F1

F1 度量：

$$F1 = \frac{2 \times P \times R}{P + R}$$

$$= \frac{2 \times TP}{\text{样例总数} + TP - TN}$$

$$\frac{1}{F1} = \frac{1}{2} \cdot \left(\frac{1}{P} + \frac{1}{R} \right)$$

若对查准率/查全率有不同偏好：

$$F_\beta = \frac{(1 + \beta^2) \times P \times R}{(\beta^2 \times P) + R}$$

$$\frac{1}{F_\beta} = \frac{1}{1 + \beta^2} \cdot \left(\frac{1}{P} + \frac{\beta^2}{R} \right)$$

$\beta > 1$ 时查全率有更大影响; $\beta < 1$ 时查准率有更大影响