解决的是近似正确的问题，研究的是学习算法：从数据中获得模型的算法

优化+学习+搜索找出最优假设

用1-ε表示趋近于1的量

有时候不需要很精准，比如预测考试分数

一般原则：奥卡姆剃刀：不要画蛇添足：若无必要，勿增实体

基本术语

- 训练：使用学习算法（决策树，神经网络......），得到模型
- 预测：新的数据样本无标记，只包含描述特征，模型应用到新数据上得到标记
- 测试数据：用于评估模型的数据
- 独立同分布：测试和训练数据类似且独立

> 重要的假设

- 泛化：训练的模型能推广到新的问题上

> 我们更关注在未知的地方做的更好，测试集准确率高才重要

- 特征信息不充分：收集不到某项数据（雾天）
- 样本信息不充分：训练数据不够多（新用户怎么推荐）

- 横向：样例=instance+label示例+标记

> 样本的真相就是标记

- 纵向：属性，特征

> 假设是在属性的基础上建立的特征到标记的关系

- 监督学习：有标记，有特征描述
- 无监督学习：只有特征输入，无标记

> 聚类，密度估计

- 先验概率：各类别出现概率
- 样例x的后验概率：x属于第i类的可能性

- 假设空间：每个点都对应了一条规则

> 机器学习是在假设空间中对假设搜索的过程

- 版本空间：与训练集一致的假设
- 归纳偏好：三个模型对训练集都100%，怎么选呢？--

> 任何有效的ML模型一定有偏好

NFL定理

> 如果某些问题上A更好，那一定有一种数据使B更好

把公式按照和谁有关重新排序

结果：所有算法一样好，不存在全局最优。方法只有和应用场景绑定才有好坏

真实目标函数. $\mathfrak{L}_a$ 的“训练集外误差”，即 $\mathfrak{L}_a$ 在训练集之外的所有样本上的误差为

$$E_{ote}(\mathfrak{L}_a|X,f)=\sum_h\sum_{\boldsymbol{x}\in\mathcal{X}-X}P(\boldsymbol{x})\,\mathbb{I}\left(h\left(\boldsymbol{x}\right)\neq f\left(\boldsymbol{x}\right)\right)P\left(h\mid X,\mathfrak{L}_a\right)\ , \qquad (1.1)$$

其中 $\mathbb{I}(\cdot)$ 是指示函数, 若 $\cdot$ 为真则取值 1, 否则取值 0.

考虑二分类问题, 且真实目标函数可以是任何函数 $\mathcal{X}\mapsto\{0,1\}$, 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

$$\begin{aligned}\sum_f E_{ote}(\mathfrak{L}_a|X,f)&=\sum_f\sum_h\sum_{\boldsymbol{x}\in\mathcal{X}-X}P(\boldsymbol{x})\,\mathbb{I}(h(\boldsymbol{x})\neq f(\boldsymbol{x}))\,P(h\mid X,\mathfrak{L}_a)\\&=\sum_{\boldsymbol{x}\in\mathcal{X}-X}P(\boldsymbol{x})\sum_h P(h\mid X,\mathfrak{L}_a)\sum_f\mathbb{I}(h(\boldsymbol{x})\neq f(\boldsymbol{x}))\\&=\sum_{\boldsymbol{x}\in\mathcal{X}-X}P(\boldsymbol{x})\sum_h P(h\mid X,\mathfrak{L}_a)\frac{1}{2}2^{|\mathcal{X}|}\\&=\frac{1}{2}2^{|\mathcal{X}|}\sum_{\boldsymbol{x}\in\mathcal{X}-X}P(\boldsymbol{x})\sum_h P(h\mid X,\mathfrak{L}_a)\\&=2^{|\mathcal{X}|-1}\sum_{\boldsymbol{x}\in\mathcal{X}-X}P(\boldsymbol{x})\cdot 1\ .\end{aligned} \qquad (1.2)$$

式(1.2)显示出, 总误差竟然与学习算法无关! 对于任意两个学习算法 $\mathfrak{L}_a$ 和 $\mathfrak{L}_b$, 我们都有

$$\sum_f E_{ote}(\mathfrak{L}_a|X,f)=\sum_f E_{ote}(\mathfrak{L}_b|X,f)\ , \qquad (1.3)$$

也就是说, 无论学习算法 $\mathfrak{L}_a$ 多聪明、学习算法 $\mathfrak{L}_b$ 多笨拙, 它们的期望性能竟

- 化简恒等式

$$\sum_f\mathbb{I}(h(\boldsymbol{x})\neq f(\boldsymbol{x}))=2^{|\mathcal{X}|-1}$$

$$\sum_h P\left(h\mid X,\mathfrak{L}_a\right)=1$$

> 选什么模型?
> 怎么配参数?

泛化误差：未来样本上

经验误差：训练样本上

过拟合：经验误差小但泛化误差大

> 所以训练到一定程度就够了，再继续就会学到不必要特征
> 不同算法有不同手段处理过拟合

```
评估方法：获得测试结果
性能度量：评估性能优劣，衡量泛化能力
假设检验：判断实质差别
```

一、评估方法

（1）留出法：一刀切

- 可分层采样：保持数据分布一致性
- 需要多次随机划分
- 2/3~4/5用于训练

（2）p次k折交叉验证法

> 常用k=10
> k=m时是留一法LOO，开销大但效果好
> 一共pk次训练

- p次随机分组
- 分成k个互斥，每次选一个作为测试
- 返回k次的均值
- 一共仅k个样本时：留一法：结果准确

评价

- k不是越大越好，开销
- 但大k会有更小的偏差
- 选择k时要最小化数据集之间的方差

5. 根据上述计算结果，我们可以发现，

$$\mathbb{E}[\bar{x}_m^*] = \mathbb{E}[\bar{x}_m] = \mu \tag{50}$$

$$var[\bar{x}_m^*] = \frac{\sigma^2}{m}[2 - \frac{1}{m}] = (2 - \frac{1}{m})var[\bar{x}_m] \approx 2var[\bar{x}_m] \tag{51}$$

另外我们如果采用 k 折交叉验证法的方式采样，类似地我们有，

$$\mathbb{E}[\bar{x}_m^{'}] = \mathbb{E}[\bar{x}_m] = \mu \tag{52}$$

$$var[\bar{x}_m^{'}] \approx \frac{k}{k-1}var[\bar{x}_m] \tag{53}$$

综上所述，虽然通过自助采样法得到的样本均值仍然是总体均值的无偏估计，但是其方差变为原来的接近两倍，而这相当于使用 2 折交叉验证采样的效果，所以一般来说，自助法采样对数据分布的改变大于交叉验证法.

（3）自助法

> 用于数据集小，难以划分时

- 缺点：改变数据集分布，引起了估计偏差（方差变为原来的近两倍）
- m次独立重复抽取
- 始终不被采到概率：

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

（4）调参

> 参数好不好影响最终性能

算法参数（超参数）：人工设定

> 算法参数设定后，要用训练集+验证集重新训练最终模型

模型参数：学习确定

- 范围+步长

> 计算开销+性能的折中方案

二、性能度量

◘ 回归(regression) 任务常用均方误差：

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

1、精度&错误率

- 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

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

2、查准率&查全率P-R图像

- FN：假的反例，应该是真的，预测成了假的
- 查准率R：所有标为正中的实际正例有多少
- 查全率R：所有的实际正例中标为正有多少
- 真正例率TPR=R
- 假正例率FPR：所有的实际反例中标为正有多少

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

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

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

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

$$\mathrm{FPR}=\frac{FP}{TN+FP}\,.$$

- 平衡点（BEP）：P=R

> 解决P-R图交叉时不易选择的问题

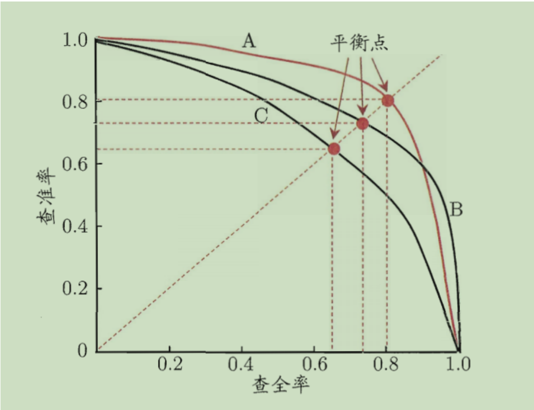
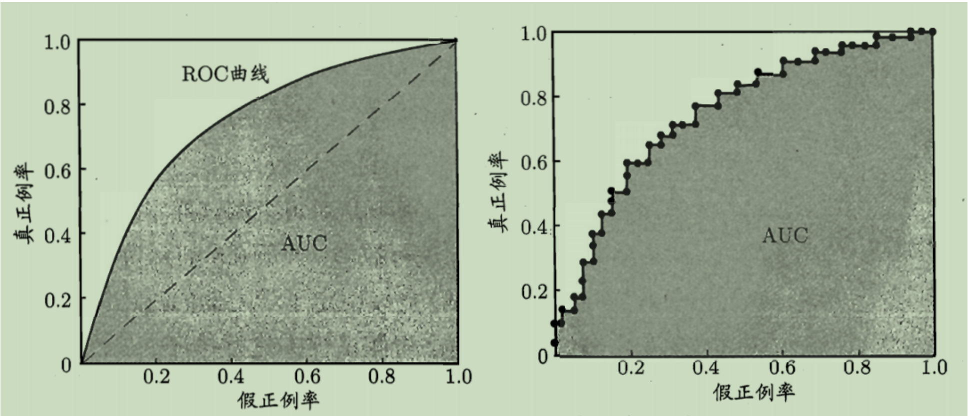


图 2.3 P-R曲线与平衡点示意图

3、ROC&AUC



(a) ROC 曲线与 AUC　(b) 基于有限样例绘制的 ROC 曲线与 AUC

- AUC：曲线下面积

相当于一堆梯形的面积和

$$\mathrm{AUC} = \frac{1}{2}\sum_{i=1}^{m-1}(x_{i+1} - x_i)\cdot(y_i + y_{i+1}) \ .$$

4、F1度量&Fβ度量

> 更常用，可调偏好
> β越大，查全率R越重要

F1 度量：

$$F1 = \frac{2 \times P \times R}{P + R} = \frac{2 \times TP}{样例总数 + TP - TN}$$

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

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

5、宏xx&微xx

> 多次测试

- 宏和微是不同的取平均方式

若能得到多个混淆矩阵：

（例如多次训练/测试的结果，多分类的两两混淆矩阵）

宏(macro-)查准率、查全率、F1

$$\text{macro-}P = \frac{1}{n}\sum_{i=1}^{n} P_i \,,$$

$$\text{macro-}R = \frac{1}{n}\sum_{i=1}^{n} R_i \,,$$

$$\text{macro-}F1 = \frac{2 \times \text{macro-}P \times \text{macro-}R}{\text{macro-}P + \text{macro-}R} \,.$$

微(micro-)查准率、查全率、F1

$$\text{micro-}P = \frac{\overline{TP}}{\overline{TP} + \overline{FP}} \,,$$

$$\text{micro-}R = \frac{\overline{TP}}{\overline{TP} + \overline{FN}} \,,$$

$$\text{micro-}F1 = \frac{2 \times \text{micro-}P \times \text{micro-}R}{\text{micro-}P + \text{micro-}R} \,.$$

6、错误非均等代价

> 将第i类样本预测为第j类样本的代价

- m+个正例，m-个反例
- D+，D-正反例集合

代价敏感(cost-sensitive)错误率：

$$E(f;D;cost) = \frac{1}{m}\left(\sum_{\boldsymbol{x}_i \in D^+} \mathbb{I}\left(f\left(\boldsymbol{x}_i\right) \neq y_i\right) \times cost_{01} + \sum_{\boldsymbol{x}_i \in D^-} \mathbb{I}\left(f\left(\boldsymbol{x}_i\right) \neq y_i\right) \times cost_{10}\right)$$

三、比较检验

> 评估结果不能用于判断优劣：

```
测试性能≠泛化性能
测试性能随测试集变化
很多ml算法本身有一定随机性
```

> 通过比较才能说某个方法好，直接说太抽象

（1）交叉验证t检验

（2）McNemar检验

算法 B	算法 A	
	正确	错误
正确	e_{00}	e_{01}
错误	e_{10}	e_{11}

若我们做的假设是两学习器性能相同，则应有 $e_{01} = e_{10}$，那么变量 $|e_{01} - e_{10}|$ 应当服从正态分布，且均值为 1，方差为 $e_{01} + e_{10}$. 因此变量

$$\tau_{\chi^2} = \frac{(|e_{01} - e_{10}| - 1)^2}{e_{01} + e_{10}} \tag{2.33}$$

服从自由度为 1 的 χ^2 分布，即标准正态分布变量的平方. 给定显著度 α，当以上变量值小于临界值 χ^2_α 时，不能拒绝假设，即认为两学习器的性能没有显著差别；否则拒绝假设，即认为两者性能有显著差别，且平均错误率较小的那个学习器性能较优. 自由度为 1 的 χ^2 检验的临界值当 $\alpha = 0.05$ 时为 3.8415，$\alpha = 0.1$ 时为 2.7055.

（3）Friedman检验

表 2.5 算法比较序值表

数据集	算法 A	算法 B	算法 C
D_1	1	2	3
D_2	1	2.5	2.5
D_3	1	2	3
D_4	1	2	3
平均序值	1	2.125	2.875

序值是每个算法在某一数据集上的好坏排序

横轴为平均序值，每个算法圆点为其平均序值，线段为临界阈值的大小

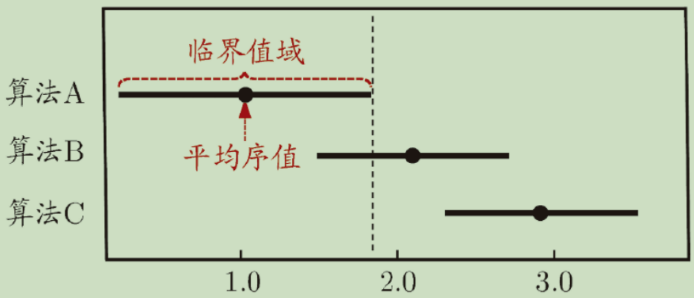


图 2.8 Friedman 检验图

若两个算法有交叠 (A 和 B)，则说明没有显著差别；
否则有显著差别 (A 和 C)，算法 A 显著优于算法 C

（4）Nemenyi检验

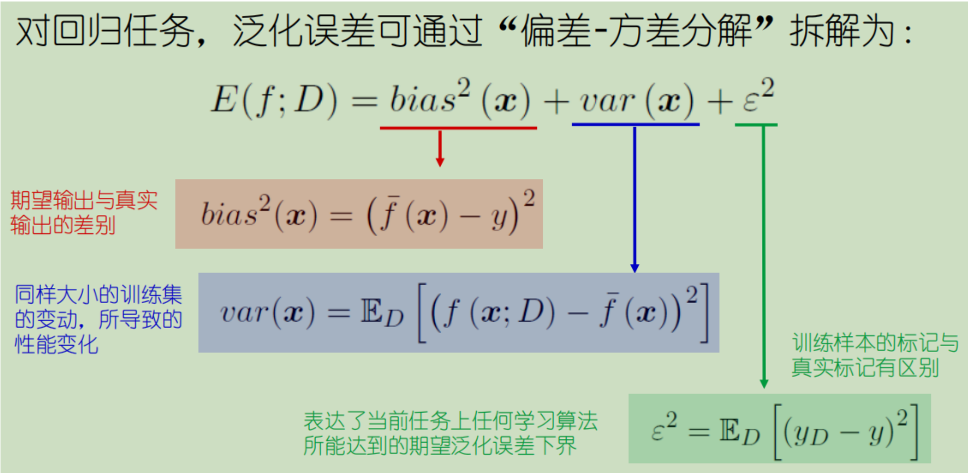
四、偏差&方差

> 泛化性能是由学习算法的能力、数据的充分性以及学习任务本身的难度所共同决定的

- 偏差bias：期望输出与真实标记的差别，刻画算法拟合能力

> 样本增多，偏差不会变

- 方差：同样大小训练集的变动导致的学习性能变化，刻画数据扰动的影响
- 噪声：当前任务任意算法所能达到的期望泛化误差下界，刻画学习问题难度



偏差-方差窘境

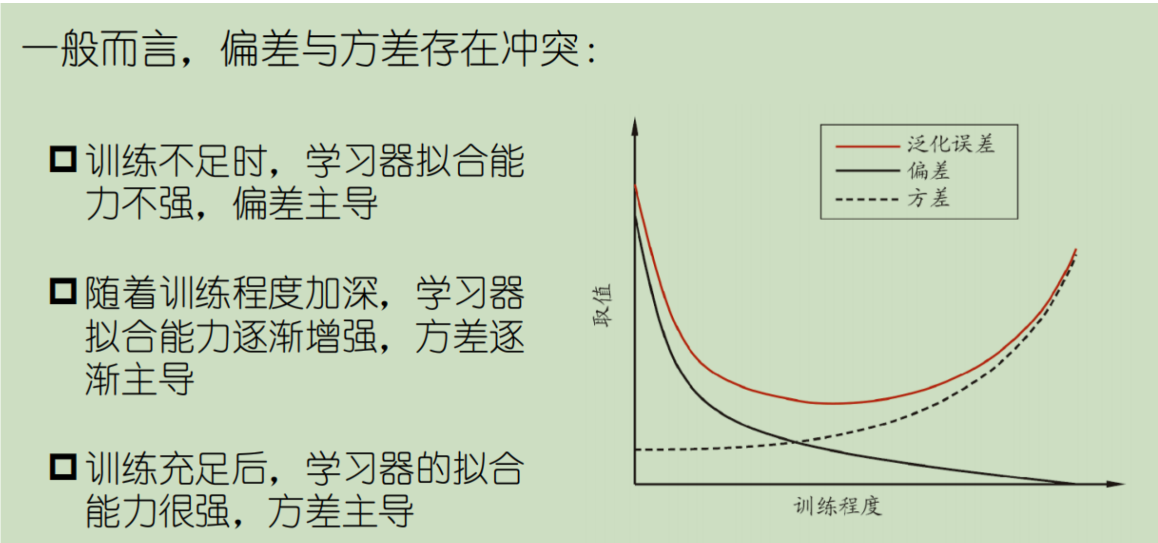


图 2.9 泛化误差与偏差、方差的关系示意图

偏差大是欠拟合（可以增加特征），方差大是过拟合