机器学习导论

第二章 模型评估与选择

经验误差与过拟合

- 错误率&误差：
  - 错误率：错分样本的占 $E = a/m$
  - 误差：样本真实输出与预测输出之间的差异
    - 训练(经验)误差：训练集上
    - 测试误差：测试集
    - 泛化误差：除训练集外所有样本

经验误差与过拟合

- 过拟合:

  学习器把训练样本学习的“太好”，将训练样本本身的特点当做所有样本的一般性质，导致泛化性能下降

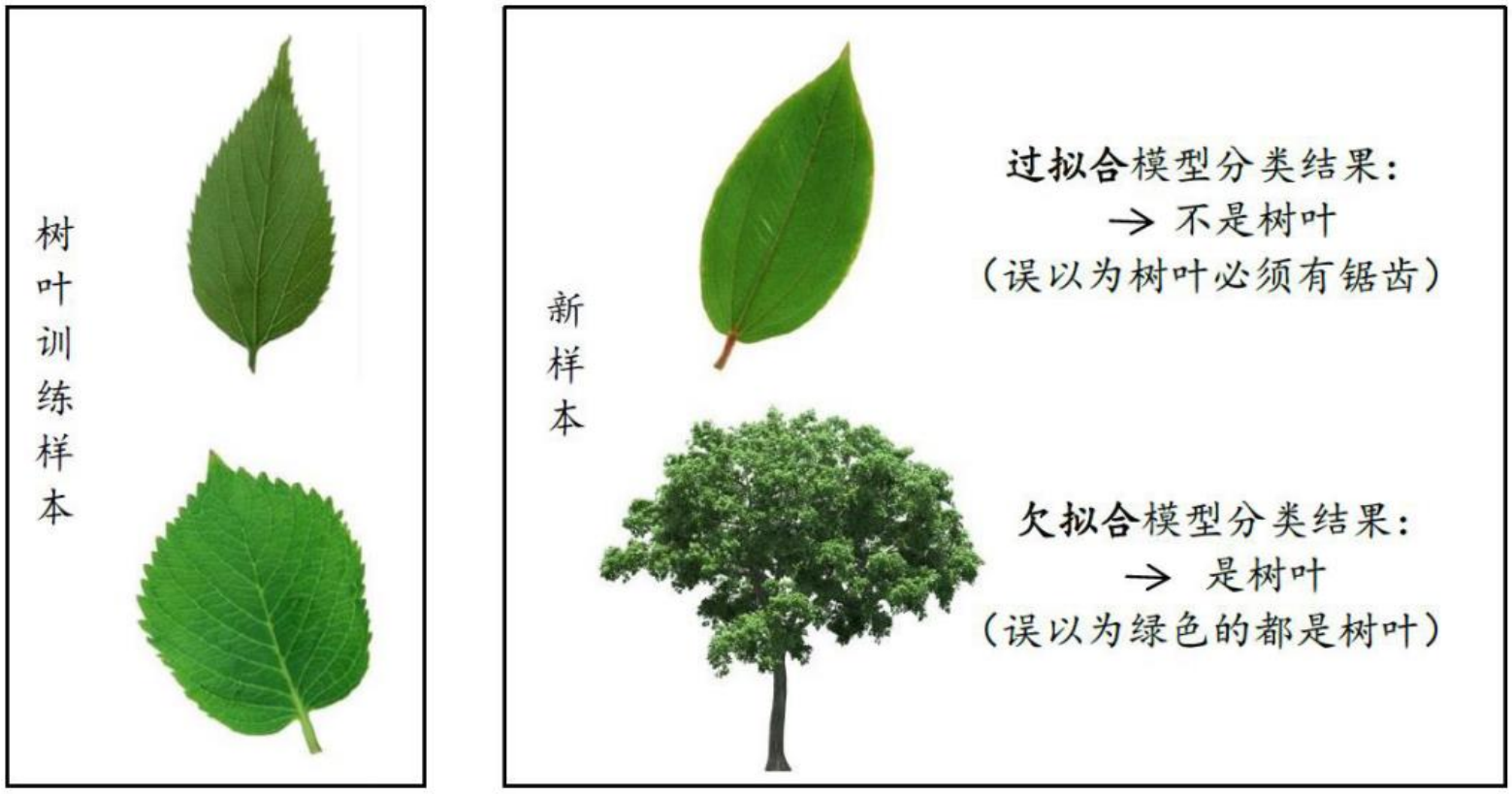
  - 优化目标加正则项
  - early stop

- 欠拟合:

  对训练样本的一般性质尚未学好

  - 决策树:拓展分支
  - 神经网络:增加训练轮数

经验误差与过拟合



过拟合、欠拟合的直观类比

评估方法

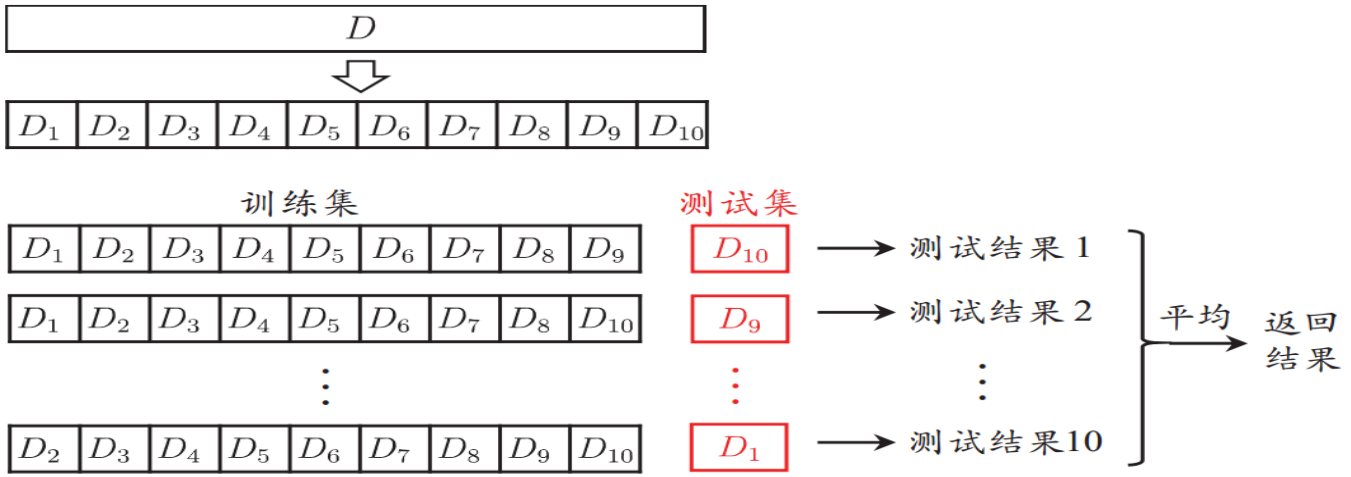
通常将包含个m 样本的数据集$D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$
拆分成训练集 S 和测试集T ：

- 留出法：
  - 直接将数据集划分为两个互斥集合
  - 训练/测试集划分要尽可能保持数据分布的一致性
  - 一般若干次随机划分、重复实验取平均值
  - 训练/测试样本比例通常为2:1~4:1

评估方法

- 交叉验证法：

将数据集分层采样划分为k个大小相似的互斥子集，每次用k-1个子集的并集作为训练集，余下的子集作为测试集，最终返回k个测试结果的均值，k最常用的取值是10.



10 折交叉验证示意图

评估方法

与留出法类似，将数据集D划分为k个子集同样存在多种划分方式，为了减小因样本划分不同而引入的差别，k折交叉验证通常随机使用不同的划分重复p次，最终的评估结果是这p次k折交叉验证结果的均值例如常见的“10次10折交叉验证”

假设数据集D包含m个样本，若令 $k = m$ ，则得到留一法：

- 不受随机样本划分方式的影响
- 结果往往比较准确
- 当数据集比较大时，计算开销难以忍受

评估方法

- 自助法：

  以自助采样法为基础，对数据集 D 有放回采样 m 次得到训练集 D'，$D \setminus D'$ 用做测试集。

  - 实际模型与预期模型都使用 m 个训练样本
  - 约有1/3的样本没在训练集中出现 ？？

性能度量

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；使用不同的性能度量往往会导致不同的评判结果

在预测任务中，给定样例集$D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$评估学习器的性能 f 也即把预测结果$f(\boldsymbol{x})$ 和真实标记比较.

回归任务最常用的性能度量是“均方误差”：

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

性能度量

对于分类任务，错误率和精度是最常用的两种性能度量：

- 错误率：分错样本占样本总数的比例
- 精度：分对样本占样本总数的比率

分类错误率

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

精度

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

性能度量

信息检索、Web搜索等场景中经常需要衡量正例被预测出来的比率或者预测出来的正例中正确的比率，此时查准率和查全率比错误率和精度更适合。

统计真实标记和预测结果的组合可以得到“混淆矩阵”

分类结果混淆矩阵

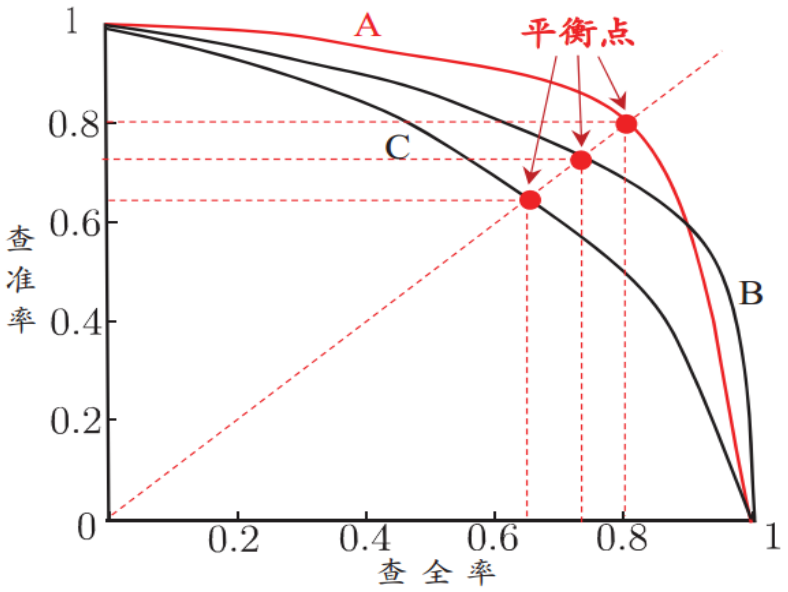
真实情况	预测结果	
	正例	反例
正例	TP (真正例)	FN (假反例)
反例	FP (假正例)	TN (真反例)

查准率 $P = \frac{TP}{TP + FP}$

查全率 $R = \frac{TP}{TP + FN}$

性能度量

根据学习器的预测结果按正例可能性大小对样例进行排序，并逐个把样本作为正例进行预测，则可以得到查准率-查全率曲线，简称“P-R曲线”



P-R曲线与平衡点示意图

平衡点是曲线上“查准率=查全率”时的取值，可用来用于度量P-R曲线有交叉的分类器性能高低

性能度量

比P-R曲线平衡点更用常用的是F1度量：

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

比F1更一般的形式 F_β ，

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

$\beta = 1$： 标准F1
$\beta > 1$： 偏重查全率(逃犯信息检索)
$\beta < 1$： 偏重查准率(商品推荐系统)

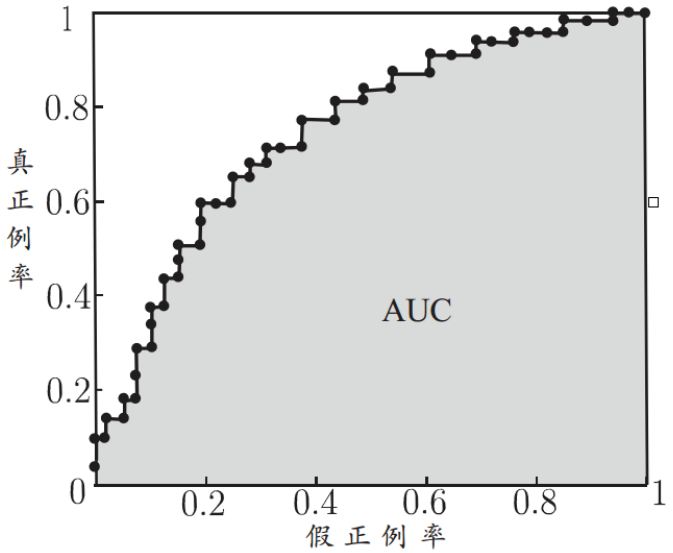
性能度量

类似P-R曲线，根据学习器的预测结果对样例排序，并逐个作为正例进行预测，以“假正例率”为横轴，“真正例率”为纵轴可得到ROC曲线，全称“受试者工作特征”.

ROC图的绘制：给定 m^+ 个正例和 m^- 个负例，根据学习器预测结果对样例进行排序，将分类阈值设为每个样例的预测值，当前标记点坐标为 (x,y)，当前若为真正例，则对应标记点的坐标为 $(x,y+\frac{1}{m^+})$；当前若为假正例，则对应标记点的坐标为 $(x+\frac{1}{m^-},y)$ ，然后用线段连接相邻点.

性能度量

若某个学习器的ROC曲线被另一个学习器的曲线“包住”，则后者性能优于前者；否则如果曲线交叉，可以根据ROC曲线下面积大小进行比较，也即AUC值.



基于有限样例绘制的 ROC 曲线与 AUC

假设ROC曲线由$\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$的点按序连接而形成，则：AUC可估算为：

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

AUC衡量了样本预测的排序质量。

代价敏感错误率

现实任务中不同类型的错误所造成的后果很可能不同，为了权衡不同类型错误所造成的不同损失，可为错误赋予“非均等代价”。

以二分类为例，可根据领域知识设定“代价矩阵”，如下表所示，其中 $cost_{ij}$ 表示将第i类样本预测为第j类样本的代价。损失程度越大，$cost_{01}$ 与 $cost_{10}$ 值的差别越大。

在非均等代价下，不再最小化错误次数，而是最小化“总体代价”，则“代价敏感”错误率相应的为：

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

性能评估

- 关于性能比较：
  - 测试性能并不等于泛化性能
  - 测试性能随着测试集的变化而变化
  - 很多机器学习算法本身有一定的随机性

直接选取相应评估方法在相应度量下比大小的方法不可取！

假设检验为学习器性能比较提供了重要依据，基于其结果我们可以推断出若在测试集上观察到学习器A比B好，则A的泛化性能是否在统计意义上优于B，以及这个结论的把握有多大。

二项检验

记泛化错误率为 ϵ，测试错误率为 $\hat{\epsilon}$，假定测试样本从样本总体分布中独立采样而来，我们可以使用“二项检验”对 $\epsilon \leq \epsilon_0$ 进行假设检验。

假设 $\epsilon \leq \epsilon_0$，若测试错误率小于

$$\bar{\epsilon} = \max \epsilon \quad \text{s.t.} \quad \sum_{i=\epsilon_0 \times m+1}^{m} \binom{m}{i} \epsilon^i (1-\epsilon)^{m-i} < \alpha$$

则在 α 的显著度下，假设不能被拒绝，也即能以 $1-\alpha$ 的置信度认为，模型的泛化错误率不大于 ϵ_0.

t检验

对应的，面对多次重复留出法或者交叉验证法进行多次训练/测试时可使用“t检验”。

假定得到了k个测试错误率，$\hat{\epsilon}_1, \hat{\epsilon}_2, \ldots, \hat{\epsilon}_k$，假设$\epsilon = \epsilon_0$对于显著度$\alpha$，若$[t_{-\alpha/2}, t_{\alpha/2}]$位于临界范围$|\mu - \epsilon_0|$内，则假设不能被拒绝，即可认为泛化错误率$\epsilon = \epsilon_0$，其置信度为$1 - \alpha$.

交叉验证t检验

现实任务中，更多时候需要对不同学习器的性能进行比较

对两个学习器A和B,若k折交叉验证得到的测试错误率分别为$\epsilon_1^A,...,\epsilon_k^A$ 和 $\epsilon_1^B,...,\epsilon_k^B$，可用k折交叉验证“成对t检验”进行比较检验。若两个学习器的性能相同，则他们使用相同的训练/测试集得到的测试错误率应相同，即 $\epsilon_i^A = \epsilon_i^B$.

偏差与方差

通过实验可以估计学习算法的泛化性能，而“偏差-方差分解”可以用来帮助解释泛化性能。偏差-方差分解试图对学习算法期望的泛华错误率进行拆解。

对测试样本x，令 y_D为 x 在数据集中的标记，y 为 x 的真实标记，$f(\boldsymbol{x};D)$ 为训练集 D 上学得模型 f 在 x 上的预测输出。以回归任务为例：学习算法的期望预期为：

$$\bar{f}(\boldsymbol{x}) = \mathbb{E}_D[f(\boldsymbol{x};D)]$$

使用样本数目相同的不同训练集产生的方差为

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

噪声为

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

偏差与方差

期望输出与真实标记的差别称为偏差，即 $bias^2(\boldsymbol{x}) = \left(\bar{f}(\boldsymbol{x}) - y\right)^2$

为便与讨论，假定噪声期望为0，也即 $\mathbb{E}_D[y_D - y] = 0$，对泛化误差分解

$$
\begin{aligned}
E(f;D) &= \mathbb{E}_D\left[\left(f(\boldsymbol{x};D) - y_D\right)^2\right] \\
&= \mathbb{E}_D\left[\left(f(\boldsymbol{x};D) - \bar{f}(\boldsymbol{x}) + \bar{f}(\boldsymbol{x}) - y_D\right)^2\right] \\
&= \mathbb{E}_D\left[\left(f(\boldsymbol{x};D) - \bar{f}(\boldsymbol{x})\right)^2\right] + \mathbb{E}_D\left[\left(\bar{f}(\boldsymbol{x}) - y_D\right)^2\right] \\
&\quad + \mathbb{E}_D\left[2\left(f(\boldsymbol{x};D) - \bar{f}(\boldsymbol{x})\right)\left(\bar{f}(\boldsymbol{x}) - y_D\right)\right] \\
&= \mathbb{E}_D\left[\left(f(\boldsymbol{x};D) - \bar{f}(\boldsymbol{x})\right)^2\right] + \mathbb{E}_D\left[\left(\bar{f}(\boldsymbol{x}) - y_D\right)^2\right]
\end{aligned}
$$

偏差与方差

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

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

$$+2\mathbb{E}_D\left[\left(\bar{f}\left(\boldsymbol{x}\right)-y\right)\left(y-y_D\right)\right]$$

又由假设中噪声期望为0，可得

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

于是： $E(f;D)=bias^2\left(\boldsymbol{x}\right)+var\left(\boldsymbol{x}\right)+\varepsilon^2$

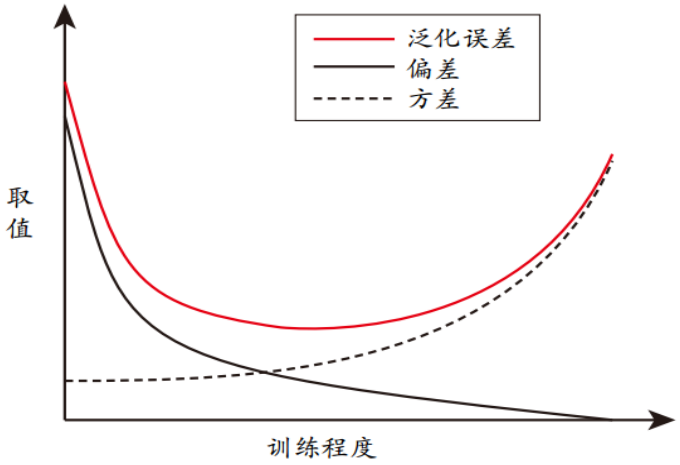
也即泛化误差可分解为偏差、方差与噪声之和。

偏差与方差

一般来说，偏差与方差是有冲突的，称为偏差-方差窘境。

如右图所示，假如我们能控制算法的训练程度：

- 在训练不足时，学习器拟合能力不强，训练数据的扰动不足以使学习器的拟合能力产生显著变化，此时偏差主导泛化错误率；
- 随着训练程度加深，学习器拟合能力逐渐增强，方差逐渐主导泛化错误率；
- 训练充足后，学习器的拟合能力非常强，训练数据的轻微扰动都会导致学习器的显著变化，若训练数据自身非全局特性被学到则会发生过拟合。



泛化误差与偏差、方差的关系示意图