# 南大周志华《机器学习》课程笔记

> Introduction：最近自学机器学习课程，注意到了南京大学周志华老师的课程。我是在学堂在线平台观看的，注意到b站上也有相应视频，但b站上并未获得授权，随时有消失的可能。
>
> 周志华老师的网络教学视频中，与其西瓜书相比确实少了一些内容。但幸运的是，缺失的内容实际上对于初学者来说并不会产生太大影响。目前这一笔记也遵循视频内容，相比西瓜书中也会有一些缺失，敬请谅解。可能以后如果有机会和时间，我会再阅读周志华老师的书籍将缺失内容补全。
>
> 一切内容敬请关注我的个人Page页面。
>
> 全系列笔记请见：click here
>
> **About Me:**点击进入我的Personal Page

## 第二章 模型评估与选择

### 泛化能力

什么模型才是"好的模型"？能够很好适用于unseen instance，例如：错误率低、精度高、召回率高等。

总体来看，我们需要一个泛化能力强的模型，但我们并没有unseen instance

泛化误差：在"未来"样本上的误差

经验误差：在训练数据上的误差，也称为"经验误差"

> - 泛化误差越小越好?
> - 经验误差越小越好?
>
> **NO!**，因为会出现过拟合/过配(overfitting)

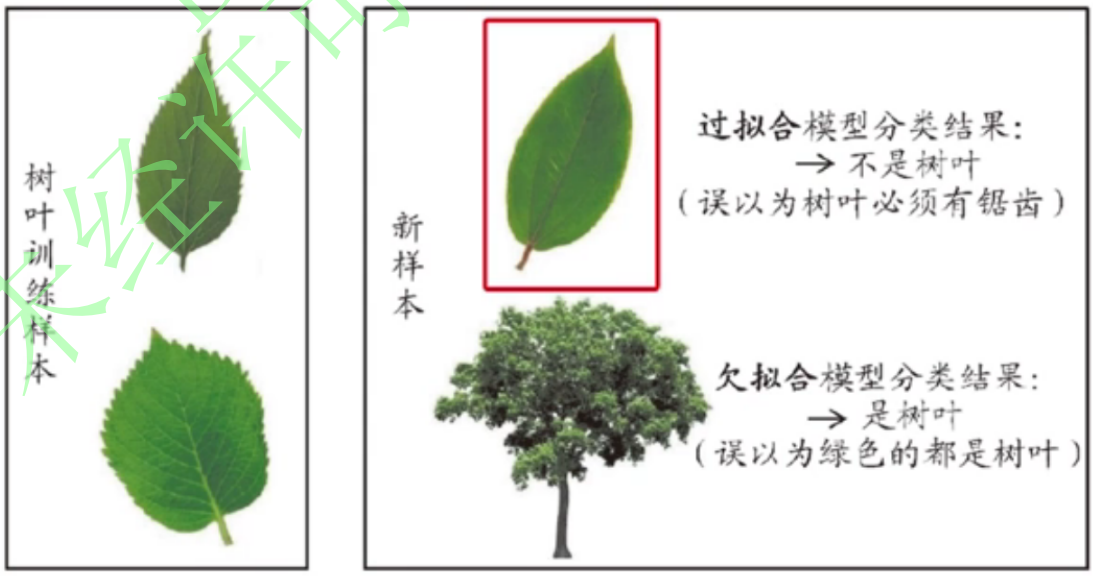



图 2.1 过拟合、欠拟合的直观类比

## 三大问题

1. 如何获得测试结果？ → 评估方法
2. 如何评估性能优劣？ → 性能度量
3. 如何判断实质差别？ → 比较检验 (防止结果表现概率优秀)

## 评估方法

关键：怎么得到测试集(Test Set)?

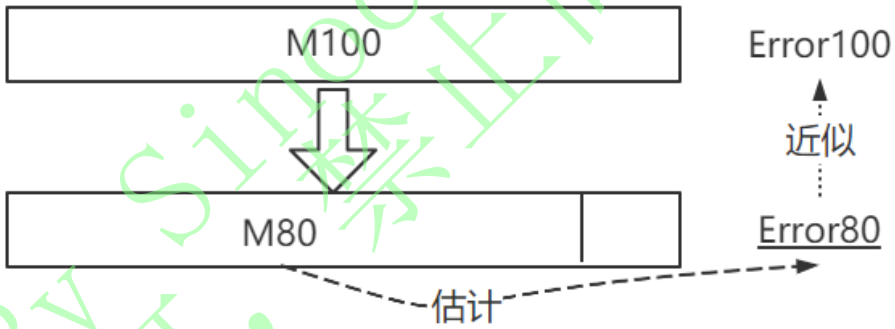
- 测试集应该和训练集”互斥“

  常见方法如下：

  - 留出法(hold-out)：将拿到的数据集分为测试集和测试集

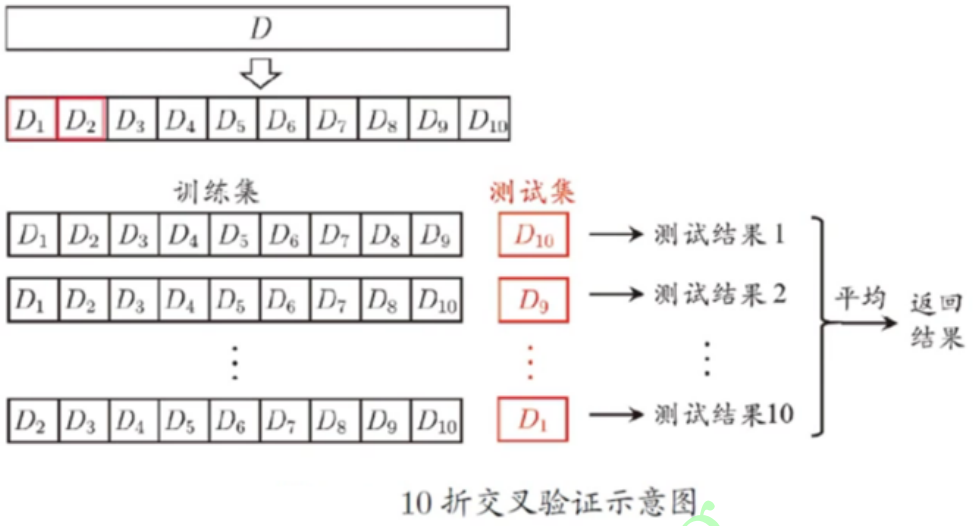
    注意：

    - 保持数据分布一致性（例如分层采样(根据类别，如好坏)）
    - 多次重复划分（例如100次随机划分）
    - 测试集不能太大，也不能太小（例如1/5 - 1/3）

    > 原因(hold-out缺陷)
    >
    > 
    > 
    >
    > 我们要的是在M100上面计算得到的Error100，但由于hold-out，我们实际是用M80得到的模型去估计Error80，再用Error80去近似Error100，如果测试集太大，那么训练模型于实际要的差异太大。
    >
    > 假设用M80训练了模型L1和L2，最后给，且最后Error上面L1表现更好，给用户的是L1吗？不是，应该是将算法再M100上运算，得到的L1’给用户。训练、测试实际上起到的是一个选择的作用。

  - 交叉验证法(cross validation)

    > hode-out是每次选择1/5 - 1/3的数据作为测试集，但仍然可能有一些数据在训练中或测试中永远没有使用过。因此可以选择用**k-折(fold)交叉验证法**。




10 折交叉验证示意图

但是，不同的划分仍有可能对数据产生影响，而这种变化可能对模型性能产生扰动。因此，可以将切分再随机做k次，这样做下来便称为k*k的交叉验证。（M90->M100）

如果每次测试集只剩余一个样本，这样称为留1法(Leave-one-out, LOO)。但这样训练的模型非常近似，但由于每次的测试样本太小，测试可能有偏差。

例如100学生，50男50女，当留出一个男生做测试，LOO认为训练集中女生多，来的是女生；当留出一个女生做测试，LOO认为训练集中男生多，来的是男生。这个模型的精度就是0.

**没有任何一种评估测试方法最好，都有其适用场景**

- 自助法(bootstrap)

思路：训练M100，留出一些样本进行测试。基于"自助采样"（bootstrap sampling，也称"有放回采样"、"可重复采样"）

用M100训练，每次从M100中取出1份，记录下来。这样有放回取样100次（即m），理论上，没有被抽中过的数据为$\lim_{m\to\infty}(1-1/m)^m = \frac{1}{e} \approx 0.368$。换言之，有大约36.8%的数据取出的集合中并未出现，可以用这部分没有出现的数据作为测试集。这种方式称为"包外估计(out-of-bag estimation)"

缺陷：改变了训练数据的分布。有时候数据量不足，或改变分布对结果影响不大时可以使用该方法。（当学习任务对数据分布的轻微变化比较鲁棒且数据量较少时）

**调参与验证集**

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

如：用线性方差进行拟合，用户提供2次还是3次，模型训练参数a,b,c等

调参过程详细：先产生若干模型，然后基于某种评估方法进行选择

参数调的好不好对性能往往对最终性能由关键影响

区别：训练集 vs 测试集 vs 验证集(validation set)

验证集——专门用于调参数（数据集分为三部分，训练集中专门用于调参的部分）

在算法参数确定后，要用 [训练集+验证集] 来重新训练最终模型

## 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)任务通常用均方误差：

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

- 错误率（每做错一次扣一分）：

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

- 精度：

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

但是上面的精度和错误率太简单了。通常来说，我们可以得到如下的一个混淆矩阵

**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

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

> 有100个西瓜，模型预测了10个是好的，10个里面有多少好瓜就是查准率；原来100个里面好瓜有20个，但是模型只给了10个，因此查全率为50%.
>
> 有时候，可能第一个算法在P上更好，第二个算法在R上更好，怎么说谁更好?

- F1度量

P=1/3,R=2/5

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

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

> 一般为了把P和R融合起来，科研使用$\sqrt{ab}$，或$\frac{a+b}{2}$，但这种a=99，b=1和a=50，b=50没有区别。采用$\frac{1}{F1}$，可以使得较小的值不会被特别忽视掉。
>
> 若对查准率/查全率有不同偏好：
>
> $$F_\beta=\frac{(1+\beta^2)\times P\times R}{(\beta^2\times P)+R}$$
>
> $$\frac{1}{F_\beta}=\frac{1}{1+\beta^2}(\frac{1}{P}+\frac{\beta^2}{R})$$
>
> 当$\beta>1$时，查全率有更大影响；当当$\beta<1$时，查准率有更大影响。

- 均方误差

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

## 比较检验

在某种度量下取得评估结果后，能否可以直接比较以评估优劣？NO！

原因1：测试性能不等于泛化性能

原因2：测试性能随着测试集的变化而变化

原因3：很多机器学习算法本身有一定的随机性

**统计假设检验(hypothesis test)为学习器性能比较提供了重要依据**

- 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

    10折，A和B学习器每次都会有一个评估指标err，获得每次二者差值。看10个差值之间的均值、方差判断谁好。
  - McNemar检验(基于列联表、卡方检验)

    类似上面【性能度量中的表2.1，关注A认为True但B认为False和A认为False但B认为True这样一个反对角线上的差异】