机器学习是目前信息技术中最激动人心的方向之一，其应用已经深入到生活的各个层面且与普通人的日常生活密切相关。本文为清华大学最新出版的《机器学习》教材的Learning Notes，书作者是南京大学周志华教授，多个大陆首位彰显其学术奢华。本篇主要介绍了该教材前两个章节的知识点以及自己一点浅陋的理解。

**1 绪论**

傍晚小街路面上沁出微雨后的湿润，和熙的细风吹来，抬头看看天边的晚霞，嗯，明天又是一个好天气。走到水果摊旁，挑了个根蒂蜷缩、敲起来声音浊响的青绿西瓜，一边满心期待着皮薄肉厚瓢甜的爽落感，一边愉快地想着，这学期狠下了工夫，基础概念弄得清清楚楚，算法作业也是信手拈来，这门课成绩一定差不了！哈哈，也希望自己这学期的machine learning课程取得一个好成绩！

**1.1 机器学习的定义**

正如我们根据过去的经验来判断明天的天气，吃货们希望从购买经验中挑选一个好瓜，那能不能让计算机帮助人类来实现这个呢？机器学习正是这样的一门学科，人的“经验”对应计算机中的“数据”，让计算机来学习这些经验数据，生成一个算法模型，在面对新的情况中，计算机便能作出有效的判断，这便是机器学习。

另一本经典教材的作者Mitchell给出了一个形式化的定义，假设：

* P：计算机程序在某任务类T上的性能。
* T：计算机程序希望实现的任务类。
* E：表示经验，即历史的数据集。

若该计算机程序通过利用经验E在任务T上获得了性能P的改善，则称该程序对E进行了学习。

**1.2 机器学习的一些基本术语**

假设我们收集了一批西瓜的数据，例如：（色泽=青绿;根蒂=蜷缩;敲声=浊响)， (色泽=乌黑;根蒂=稍蜷;敲声=沉闷)， (色泽=浅自;根蒂=硬挺;敲声=清脆)……每对括号内是一个西瓜的记录，定义：

* 所有记录的集合为：数据集。
* 每一条记录为：一个实例（instance）或样本（sample）。
* 例如：色泽或敲声，单个的特点为特征（feature）或属性（attribute）。
* 对于一条记录，如果在坐标轴上表示，每个西瓜都可以用坐标轴中的一个点表示，一个点也是一个向量，例如（青绿，蜷缩，浊响），即每个西瓜为：一个特征向量（feature vector）。
* 一个样本的特征数为：维数（dimensionality），该西瓜的例子维数为3，当维数非常大时，也就是现在说的“维数灾难”。

计算机程序学习经验数据生成算法模型的过程中，每一条记录称为一个“训练样本”，同时在训练好模型后，我们希望使用新的样本来测试模型的效果，则每一个新的样本称为一个“测试样本”。定义：

* 所有训练样本的集合为：训练集（trainning set），[特殊]。
* 所有测试样本的集合为：测试集（test set），[一般]。
* 机器学习出来的模型适用于新样本的能力为：泛化能力（generalization），即从特殊到一般。

西瓜的例子中，我们是想计算机通过学习西瓜的特征数据，训练出一个决策模型，来判断一个新的西瓜是否是好瓜。可以得知我们预测的是：西瓜是好是坏，即好瓜与差瓜两种，是离散值。同样地，也有通过历年的人口数据，来预测未来的人口数量，人口数量则是连续值。定义：

* 预测值为离散值的问题为：分类（classification）。
* 预测值为连续值的问题为：回归（regression）。

我们预测西瓜是否是好瓜的过程中，很明显对于训练集中的西瓜，我们事先已经知道了该瓜是否是好瓜，学习器通过学习这些好瓜或差瓜的特征，从而总结出规律，即训练集中的西瓜我们都做了标记，称为标记信息。但也有没有标记信息的情形，例如：我们想将一堆西瓜根据特征分成两个小堆，使得某一堆的西瓜尽可能相似，即都是好瓜或差瓜，对于这种问题，我们事先并不知道西瓜的好坏，样本没有标记信息。定义：

* 训练数据有标记信息的学习任务为：监督学习（supervised learning），容易知道上面所描述的分类和回归都是监督学习的范畴。
* 训练数据没有标记信息的学习任务为：无监督学习（unsupervised learning），常见的有聚类和关联规则。

**2 模型的评估与选择**

**2.1 误差与过拟合**

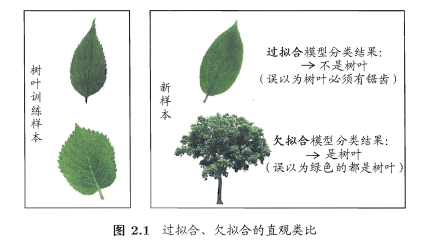
我们将学习器对样本的实际预测结果与样本的真实值之间的差异成为：误差（error）。定义：

* 在训练集上的误差称为训练误差（training error）或经验误差（empirical error）。
* 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（generalization error）。

显然，我们希望得到的是在新样本上表现得很好的学习器，即泛化误差小的学习器。因此，我们应该让学习器尽可能地从训练集中学出普适性的“一般特征”，这样在遇到新样本时才能做出正确的判别。然而，当学习器把训练集学得“太好”的时候，即把一些训练样本的自身特点当做了普遍特征；同时也有学习能力不足的情况，即训练集的基本特征都没有学习出来。我们定义：

* 学习能力过强，以至于把训练样本所包含的不太一般的特性都学到了，称为：过拟合（overfitting）。
* 学习能太差，训练样本的一般性质尚未学好，称为：欠拟合（underfitting）。

可以得知：在过拟合问题中，训练误差十分小，但测试误差教大；在欠拟合问题中，训练误差和测试误差都比较大。目前，欠拟合问题比较容易克服，例如增加迭代次数等，但过拟合问题还没有十分好的解决方案，过拟合是机器学习面临的关键障碍。

[](https://camo.githubusercontent.com/3c5a03bf3c29a9c1926ce013d3ac938e93d87ac9/68747470733a2f2f692e6c6f6c692e6e65742f323031382f31302f31372f356263373138313137323939362e706e67)

**2.2 评估方法**

在现实任务中，我们往往有多种算法可供选择，那么我们应该选择哪一个算法才是最适合的呢？如上所述，我们希望得到的是泛化误差小的学习器，理想的解决方案是对模型的泛化误差进行评估，然后选择泛化误差最小的那个学习器。但是，泛化误差指的是模型在所有新样本上的适用能力，我们无法直接获得泛化误差。

因此，通常我们采用一个“测试集”来测试学习器对新样本的判别能力，然后以“测试集”上的“测试误差”作为“泛化误差”的近似。显然：我们选取的测试集应尽可能与训练集互斥，下面用一个小故事来解释why：

假设老师出了10 道习题供同学们练习，考试时老师又用同样的这10道题作为试题，可能有的童鞋只会做这10 道题却能得高分，很明显：这个考试成绩并不能有效地反映出真实水平。回到我们的问题上来，我们希望得到泛化性能好的模型，好比希望同学们课程学得好并获得了对所学知识"举一反三"的能力；训练样本相当于给同学们练习的习题，测试过程则相当于考试。显然，若测试样本被用作训练了，则得到的将是过于"乐观"的估计结果。

**2.3 训练集与测试集的划分方法**

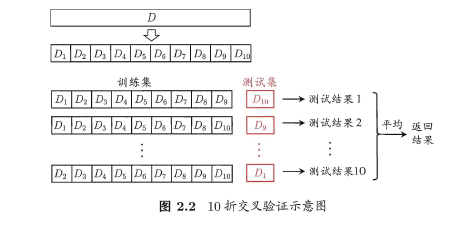
如上所述：我们希望用一个“测试集”的“测试误差”来作为“泛化误差”的近似，因此我们需要对初始数据集进行有效划分，划分出互斥的“训练集”和“测试集”。下面介绍几种常用的划分方法：

**2.3.1 留出法**

将数据集D划分为两个互斥的集合，一个作为训练集S，一个作为测试集T，满足D=S∪T且S∩T=∅，常见的划分为：大约2/3-4/5的样本用作训练，剩下的用作测试。需要注意的是：训练/测试集的划分要尽可能保持数据分布的一致性，以避免由于分布的差异引入额外的偏差，常见的做法是采取分层抽样。同时，由于划分的随机性，单次的留出法结果往往不够稳定，一般要采用若干次随机划分，重复实验取平均值的做法。

**2.3.2 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=D1∪D2∪...∪Dk，Di∩Dj=∅（i≠j），同样地尽可能保持数据分布的一致性，即采用分层抽样的方法获得这些子集。交叉验证法的思想是：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。交叉验证法也称“k折交叉验证”，k最常用的取值是10，下图给出了10折交叉验证的示意图。

[](https://camo.githubusercontent.com/bc0633094f7c687bb591366fc0a0a65fb489b789/68747470733a2f2f692e6c6f6c692e6e65742f323031382f31302f31372f356263373138313135643232342e706e67)

与留出法类似，将数据集D划分为K个子集的过程具有随机性，因此K折交叉验证通常也要重复p次，称为p次k折交叉验证，常见的是10次10折交叉验证，即进行了100次训练/测试。特殊地当划分的k个子集的每个子集中只有一个样本时，称为“留一法”，显然，留一法的评估结果比较准确，但对计算机的消耗也是巨大的。

**2.3.3 自助法**

我们希望评估的是用整个D训练出的模型。但在留出法和交叉验证法中，由于保留了一部分样本用于测试，因此实际评估的模型所使用的训练集比D小，这必然会引入一些因训练样本规模不同而导致的估计偏差。留一法受训练样本规模变化的影响较小，但计算复杂度又太高了。“自助法”正是解决了这样的问题。

自助法的基本思想是：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'。可以得知在m次采样中，样本始终不被采到的概率取极限为：

[https://camo.githubusercontent.com/e597257a4848460d51344dc8e5e996930dc6ffb2/68747470733a2f2f692e6c6f6c692e6e65742f323031382f31302f31372f356263373138313132343664642e706e67](https://camo.githubusercontent.com/e597257a4848460d51344dc8e5e996930dc6ffb2/68747470733a2f2f692e6c6f6c692e6e65742f323031382f31302f31372f356263373138313132343664642e706e67)

这样，通过自助采样，初始样本集D中大约有36.8%的样本没有出现在D'中，于是可以将D'作为训练集，D-D'作为测试集。自助法在数据集较小，难以有效划分训练集/测试集时很有用，但由于自助法产生的数据集（随机抽样）改变了初始数据集的分布，因此引入了估计偏差。在初始数据集足够时，留出法和交叉验证法更加常用。

**2.4 调参**

大多数学习算法都有些参数(parameter) 需要设定，参数配置不同，学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" (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*5*5= 125 个模型需考察，由此可见：拿下一个参数（即经验值）对于算法人员来说是有多么的happy。

最后需要注意的是：当选定好模型和调参完成后，我们需要使用初始的数据集D重新训练模型，即让最初划分出来用于评估的测试集也被模型学习，增强模型的学习效果。用上面考试的例子来比喻：就像高中时大家每次考试完，要将考卷的题目消化掉（大多数题目都还是之前没有见过的吧？），这样即使考差了也能开心的玩耍了~。