第 5 章 机器学习流程、评价和性能界

CHAPTER 5

正如第 1 章中“没有免费的午餐”定理所述，没有一种机器学习模型是对所有问题普遍适用的，正因为如此，机器学习领域中提出了众多的不同模型。在第 4 章介绍了基本的回归和分类模型，在第 6 章之后仍将继续介绍多种常用的机器学习模型。在这样一个节点，偏离一下机器学习模型和学习算法的介绍，以一章的篇幅集中介绍机器学习中训练和性能评价的一些基本问题，这是一个很大的问题，本章的介绍仅仅是入门和概要性质的。

当用选定模型解决一类问题时，对模型性能的理想描述是期望风险，即从完整的统计意义上刻画模型相对于目标的偏差。但在机器学习领域，缺乏对目标完整的概率描述，因此无法获得期望风险，需要用从有限数据中学习模型的方法，评价准则也以经验风险代替期望风险。由于数据集的代表能力有限，以经验风险最优确定的模型对真实目标的总体表达能力如何？即泛化性能如何？这是一个非常关键的问题。


第 10 集
微课视频

泛化性能好是一个机器学习模型可用的基本要求，因此必须要对泛化性能进行评价。一种比较实际的评价泛化性能的方法是通过数据集进行测试，将数据集划分为训练集和测试集，用训练集学习模型，在测试集上近似估计其泛化性能；第二种评价方法是给出理论上的泛化界并研究泛化误差与数据集规模的关系，这是机器学习理论讨论的基本问题。遗憾的是，目前这两种方法之间仍存在鸿沟。利用机器学习理论，对于在要求的泛化误差下给出的样本规模并不能很精确地指导许多实际机器学习模型的训练，逾越这道鸿沟仍需艰难的研究工作。

本章由两部分组成：第 1 部分包括前两节，首先给出机器学习流程的一个概要讨论，然后讨论如何利用实际数据集有效地评价一个机器学习模型；第 2 部分由后两节组成，讨论了机器学习理论中的一些基本概念和结论，以期帮助读者对机器学习理论有一个基本的了解。

5.1 机器学习流程

在 1.1 节的概述中，给出了机器学习的一个化简流程，用于说明机器学习过程的基本元素，在学习了前 4 章的概述及基本模型与算法，对机器学习有了一些基础后，这里再对机器学习的流程给出一个更细致的框图，需要注意的是，机器学习流程中的一些因素是与应用环境密切相关的，超出本书的范围，本节给出机器学习系统的总体结构。

5.1.1 机器学习基本流程

在图 1.1.1 基础上进一步细化，给出更详细的机器学习流程图，如图 5.1.1 所示。

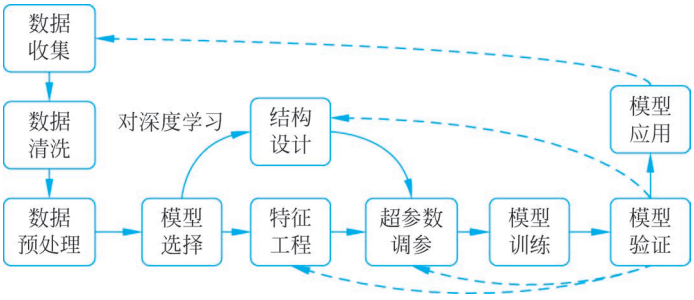


图 5.1.1 机器学习流程

由于机器学习的模型的训练方法主要是数据驱动的，因此数据收集是构成一个机器学习系统的第一步。对于许多较为通用的领域，已存在大量数据集可供选用，例如：语音、图像、视频和文字等通用领域，有各种公开的大规模或中规模数据集。以图像为例，既有综合类图像数据集，也有一些专用数据集，如人脸、手写字体、建筑物等。这些数据集可用于各种不同目的的模型训练和验证。也有一些专业类型的数据集，可在行业内部或公开使用，例如：用于医学辅助诊疗的医学图像数据集、用于合成孔径雷达图像目标分类的数据集等，这种专业数据集也越来越丰富。在一些特定应用中，也可以收集特定的数据集。在一些专业应用中，可将大规模公开数据集和小规模专业数据集结合，利用公开数据集预训练一个表征模型，通过小规模专业数据集细化模型参数，获得最终可用的系统。关于数据收集需了解其来源的多样性，但不作为机器学习教材的核心内容。

数据清洗和数据预处理可以合并统称为数据预处理，这一部分是在将数据用于选定模型训练前，对数据集做适应性处理，例如剔除不合格数据，将数据集格式规范化等，对这一部分，5.1.2 节再做专门介绍。

模型选择是在众多的机器学习模型中选择一种来实现任务。机器学习有众多模型，第 1 章介绍了各种模型的分类。模型选择首先要选定一个大的类型：例如要完成对图像中目标的分类，就选择分类模型；若是进行股票指数的预测，则选择回归模型。在选择大类型后，还要选择一种具体的模型实现其功能。例如，要进行分类，可以选择具体模型，如：逻辑回归、朴素贝叶斯、支持向量机、决策树、梯度提升树、CNN 等。对于模型的选择要根据任务需求、数据规模、模型的特点等综合考虑。要选择好模型，最重要的是了解各种模型的性质和能力，本书在第 4 章介绍了基本的回归和分类模型，后续第 6～16 章继续介绍大量不同模型，掌握这些知识是选择模型的基础。

在选择了模型后，根据模型是否为深度学习模型，分为两条支路。若选择传统的模型，例如支持向量机等，如果输入向量$\tilde{\boldsymbol{x}}$ 的维度特别高(例如一幅图像或一段语音)，则一般不直接将原始输入作为模型的输入，而是从$\tilde{\boldsymbol{x}}$ 中抽取一组特征向量 $\boldsymbol{x}$ 作为模型的输入，一般 $\boldsymbol{x}$ 的维度比$\tilde{\boldsymbol{x}}$ 低得多，这一步称为特征工程，$\boldsymbol{x}$ 应能表示$\tilde{\boldsymbol{x}}$ 的主要信息且具有更清晰的含义。5.1.4 节对特征工程再做更详细一些的介绍。

若选择深度学习模型，则一般不需要再抽取特征向量，可将高维输入直接送入模型，由模型自动抽取多层次的特征表示。但由于深度学习模型以深层神经网络为主，其中结构的灵活性非常高，需要对模型结构进行进一步设计。例如，选择什么结构的神经网络？全连接结构、CNN 结构、RNN 结构还是 Transformer 结构？如果选择了 CNN 结构，需确定要多少层？每层多少个卷积核？卷积核长度是多少？多少卷积层后加一池化层？选择什么非线性激活函数？最后需要不需要全连接层？也就是说，根据需要设计合适的深度网络结构，经常需要在训练结束后，根据性能反馈再调整结构。人们也提出了自动搜索最佳结构的方法。因此，对于深度学习模型来说，结构设

计是重要的一步。本书第 9～11 章介绍多种深层网络模型，第 15 章介绍深度强化学习，第 16 章介绍深度生成模型，这些内容可以为读者进行结构选择和设计打下基础。

之后的“超参数调整”和“模型训练”是紧密结合、反复迭代的两个模块。由于各种情况引出对数据集划分的不同，以及迭代方式有多种，这一部分放在 5.1.3 节中做更详细的介绍。其中，模型训练算法也是课程的核心：机器学习算法。本书第 4 章和第 6～16 章主要介绍各种模型的结构，以及在此基础上怎样通过学习算法确定模型结构和参数。

不管学习过程或训练过程中怎样划分数据集，是将数据集划分为训练集和测试集，还是划分为训练集、验证集(Validation Set)和测试集，总归是要留出一部分数据作为测试集，来评价最后训练完成的模型性能，这一步称为模型验证。测试集不参与模型训练，为的是留出相对独立的数据集进行模型的验证，这个验证过程可以部分地测试模型的泛化性能。

若训练用的目标函数是一种误差函数，可以将其用在计算训练集和测试集误差，评价训练后模型泛化性能。若训练集的误差大，测试集的误差也大，说明模型是欠拟合的，应该选用更复杂的模型重新训练和验证。若训练集的误差很小，测试集的误差大，说明模型是过拟合的，需要重新训练模型，重新训练时，可选择更简单的模型，或采用更强的正则化条件，或增加数据集规模(若没有条件直接扩大数据集，可选择适当的增广方法)，重新训练后，再次进行模型验证。若模型在训练集和数据集上误差均小，且达到设计要求，则可以将模型部署应用。

由于大多数模型的复杂性使得解析解不存在，训练算法大多是迭代式算法，需要求目标函数对模型参数的导数，因此目标函数的选择一方面要反映模型性能，另一方面还要对模型参数连续可导，因此许多目标函数是性能评价的一种代理函数，而不是直接反映与应用需求对应的性能。例如，分类模型训练中常用的目标函数是交叉熵，而最直接反映分类性能的评价是误分类率，但误分类率是模型参数的不连续函数，无法直接用于优化。尽管交叉熵是对误分类率的一种好的替代目标函数，但毕竟有区别，故模型评价这一步除了如上所述通过目标函数判断泛化性能外，还常常根据应用需求，用更实际的评价函数来评估模型的性能，若模型达不到实际性能的要求，则可以通过改变模型选择和结构后重新训练，故模型评价的结果(尤其不满足要求时)需要反馈回前面的模块。5.2 节进一步介绍一些常用的实际评价函数。

流程中的最后一个模块是模型应用。当训练完成的模型经过了性能的验证后，可部署应用。对于大多数机器学习系统，其训练过程和应用时的推断或预测过程是不平衡的，尤其深度学习模型，其训练过程极为耗时耗力，但其部署应用时，运算复杂度不高，甚至可嵌入移动设备中。从一个完整循环的角度来讲，部署应用是最后一步，但从持续发展的角度来讲，一个有意义的应用是不断改进和完善的，应用的效果可反馈给设计者，设计者也可能不断积累更多数据，计算资源不断提高，可支持系统的不断更新和性能的提高。对于一些特殊的在线应用，例如推荐系统，可将用户反馈用于在线改进系统的性能。

本小节给出了机器学习流程的一个概述，其中模型结构和学习算法是本书的主要内容，流程中的其他元素在本章其他小节给出进一步的概要介绍。本章后半部分还对机器学习的理论给出一些概要介绍，使得读者对机器学习有一个更全面的认识。第 4 章已经介绍了基本的回归和分类的模型和算法，本书后续各章用于介绍各种更复杂的模型和算法。

5.1.2　数据清洗和数据预处理

数据清洗和数据预处理也可归并为数据预处理，但若细化地考虑，两者还是有所不同。数据清洗主要是剔除数据集中不合格的样本，而数据预处理则是对样本做简单加工，使之更适合于直

接输入模型中。

在数据预处理环节，尽量不做复杂运算，若特征向量是高维的，一般每一维单独处理，故在本小节中，样本集使用标量形式$\{x_n, y_n\}_{n=1}^{N}$，其表示特征向量的其中1维，每一维采用相同的处理方法。

数据清洗主要有两方面工作：缺失特征的填补和异常数据的剔除。

样本集中的部分样本可能有缺失的特征分量，以有监督学习的分类为例，既然这里用标量特征做说明，所谓特征缺失，即对于样本n，其x_n取值缺失，若需要对其进行填补，有一些常用的填补方法。设分类类型共有C类，按标注y_n的取值将样本集分为C个子集$D_k=\{x_n \mid y_n=k\}$，$k=1,2,\cdots,C$，假设每一类服从高斯分布，计算各子样本集的均值和方差：μ_k, σ_k^2。如果一个样本n对应第k类，其x_n缺失了，有一些常用的填补方法，这里给出3种实例。(1)取$x_n=\mu_k$；(2)取$x_n \sim N(\mu_k, \sigma_k^2)$，即通过其PDF采样获得一个随机值赋予$x_n$；(3)若不希望做概率估计且样本集规模较小，可取$x_n$为$D_k$的中值。对于回归情况，可将所有样本只构成一个数据集进行处理，也可将标注取值划分成几个区间，相应构成几个子集进行处理。

若特征向量是高维的，每一维均可这样处理，但若一个样本缺失的分量较多，则删除此样本是更好的选择。如果一个样本的标注缺失，对于有监督学习来讲，该样本作用不大，可删去。对于一个数据集，很难预先确定哪种填补方式更好，可以通过实验进行测试比较。还有一些填补方式与算法流程结合，例如，在决策树中，可不去预测填补的值，而是以概率加权的方式将有缺失样本分到不同的子集中，关于这种方法请参考7.3节。

异常值检测和异常样本剔除是数据清洗的另一个常见任务。异常值是指偏离了正常取值区间的值，异常值对一些机器学习算法影响较大，例如，建立在平方误差目标函数上的回归模型，对异常值很敏感。对于取值连续的变量(输入分量或回归的标注)，若设其服从高斯分布，均值和方差为μ, σ^2(标准差为σ)，则其取值与均值之差$|x-\mu|$大于2σ的概率小于0.05，$|x-\mu|$大于3σ的概率仅为0.003，可见，正常取值偏离均值μ的范围受限，故可定义一个样本x_n的Z-得分为

$$Z_n=\frac{|x_n-\mu|}{\sigma} \tag{5.1.1}$$

一种判断异常值的办法是给出一个合理的门限T，若样本x_n满足$Z_n>T$，则判断其为异常值。若一个样本被判断为异常值，常用的处理方法是将其删除。

当样本分量取值逼近高斯分布时，Z-得分方法判断异常值是有效的，但是，当输入是多峰值的PDF时，该方法不再有效。在这种情况下，输入特征向量常趋于一种聚类分布，可以利用第12章的聚类算法剔除异常值。即将数据集除去标注后，通过无监督学习的方法形成K个聚类，当一个样本偏离各聚类中心均超过一个门限时，将其判断为异常值。有关聚类的算法在第12章介绍。

当对数据集进行了数据清洗后，为了使每个样本适应模型的要求，往往还需要进行一些预处理。许多机器学习模型在归一化的输入情况下，收敛和性能更好。这里的归一化有两种常见形式，一是取值范围归一化，二是概率分布归一化。取值范围归一化指每个输入x_n取值在固定范围，例如[0,1]或[−1,1]，这里以前者为例进行说明。对于数据集，记录其最小值和最大值分别为：$x_{\min}, x_{\max}$，则每个样本预处理为

$$x_n \leftarrow \frac{x_n-x_{\min}}{x_{\max}-x_{\min}} \tag{5.1.2}$$

对于一些规范的对象，$x_{\min}, x_{\max}$是可预定的，不必从样本集中搜索，例如8比特的图像数据，$x_{\min}=0, x_{\max}=255$，14比特的高保真音乐，$x_{\min}=-2^{13}, x_{\max}=2^{13}-1$。

概率分布归一化是将样本集归一化为均值为0，方差为1的分布，可由样本集估计均值和方

差：μ,σ^2，则每个样本做如下归一化。

$$x_n \leftarrow \frac{x_n-\mu}{\sigma} \tag{5.1.3}$$

有一些算法要求特殊的归一化，如在 4.2 节介绍的 Lasso 算法中，要求做归一化为 $\Sigma=\sum_{n=1}^{N}x_n^2$，$x_n \leftarrow x_n/\Sigma$。

在模型训练过程中，确定的归一化参数，如 $x_{\min}$，$x_{\max}$，μ，σ 等，在模型应用时，将输入特征向量用同样的参数做归一化处理。

由于输入特征向量可能维度很高，目前介绍的归一化只在各分量独立完成，使得各分量分布一致，但并不去除分量之间的相关性。若要通过归一化将高维特征向量归一化为均值为 0，协方差矩阵为单位矩阵的去相关向量，则需要高维特征分析，具体算法在 13.2 节介绍，这种预处理在大规模数据集上实现的运算复杂度太高，故在深度学习的训练中，主要采用的是各分量单独预处理的方法。

对于监督学习，数据集的标注也需要预处理，尤其针对分类的标注。很多分类问题的标注使用了文字表示标注，在将数据集用于训练时，首先将文字标注转换成数字标注，若已经是数字标注的，要转换成模型需要的格式。例如，2 分类情况，若标注用文字为“是”或“非”，需要根据模型的要求，转换为“1、0”或“1、−1”。对于多分类，大多数算法要求的标注是 K 维独热编码，需要将文字标注或单一数字标注转换为 K 维向量编码，例如一个 4 种类型的分类问题，原数据集用{1,2,3,4}标注 4 种类型，需转换为独热编码{1000,0100,0010,0001}。

一般来讲，对于目前的机器学习模型，合理的预处理可提升训练效率和性能。

5.1.3　模型的训练、验证与测试

在机器学习的训练和验证时，最简单方法的是将数据集分为训练集和测试集，更多情况下，对数据集的划分更复杂。本小节结合实际 ML 系统的学习过程，对数据的划分和作用做一些更深入的讨论。

在 ML 的许多模型中，存在一些称为超参数的量，超参数是不能直接通过训练过程确定得到的。例如多项式拟合的阶 M，KNN 的参数 K，或正则项的控制参数 λ。可以通过学习理论或贝叶斯框架下的学习确定这些超参数，但目前在实际中更常用的是通过验证过程确定。

在最简单的情况下，不需要确定超参数，数据集仍划分为训练集和测试集，或两个集合独立地产生自同一个数据生成分布，训练集训练模型，通过测试误差近似评价泛化性能。

更复杂的情况下可将数据集划分为三个集合，训练集、验证集和测试集。若数据集数据量充分，可以直接按一定比例划分三个集合，例如训练集占 80%、验证集占 10%和测试集占 10%，各集合的比例可根据数据集的总量做适当调整，数据集划分的示意图如图 5.1.2(a)所示。在一般的学习过程中，在超参数的取值空间内，按一定方式(等间隔均匀取值或随机取值)取一个(或一组，复杂模型可能有多个超参数)超参数值，用这个确定的超参数，通过训练集训练模型，将训练得到的模型用于验证集，计算验证集误差。取不同的超参数，重复这个过程，最后确定效果最好的超参数以及对应的模型。然后用测试集测试性能，计算测试误差，估计泛化性能。若测试性能达不到要求，还可能回到原点，选择不同的模型，重复以上过程，直到达到要求或在可能选择的模型中取得最好结果。在整个过程中测试集是不参与学习过程的，这样才能够得到可信的评估模型的泛化能力。

在一些情况下数据集规模较小，若固定的分成三个集合，因每个集合数据量小使得训练过程和验证过程都缺乏可靠性。这种情况可采用交叉验证(Cross Validation)方法。数据集仍划分为测试集和训练集，测试集留作最后的测试用。将训练集分为 K 折(K Folds)，用于训练和验证，对于一组给出的超参数，做多轮训练，每次训练留出一折作为验证集，其余作为训练集，进行一次训练和验证，然后循环操作，过程如图 5.1.2(b)所示。做完一个循环，将每次验证集的误差做平均，作为验证误差。选择一组新的超参数重复该过程。直到全部需要实验的超参数取值完成后，比较所有超参数取值下的验证误差，确定超参数的值，其后再用全部训练集样本训练出模型。将以上学习过程确定的模型用于测试集去计算测试误差，评价是否达到目标。

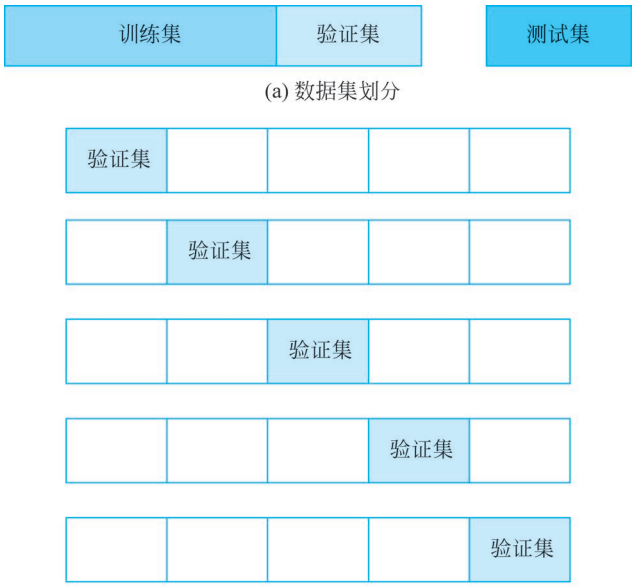


图 5.1.2　用于训练和测试的数据集划分

在数据集样本相当匮乏的情况下，以上交叉验证可取其极限情况，每一轮只留一个样本作为验证集，称为留一交叉验证(Leave-one out Cross Validation，LOOCV)。

以上介绍了用数据集获得一个 ML 模型的基本方法。实践中可能还有各种灵活的组合方式。第 4 章以及后续章节介绍的各种算法在实际应用时，一般用以上的某一种方式完成训练和测试过程。

5.1.4　特征工程

在图 5.1.1 中，特征工程是可跳过的模块。在多数深度学习模型的训练时，可直接输入高维数据，一般不需要特征工程模块。在传统机器学习领域，当输入向量维度较低时，一般也不需要做进一步的特征选择，即使各分量存在冗余，通过正则化方法也可有效改善。但当传统模型面对很高维输入向量，其或存在明显的冗余，或当高维输入中包含的关键信息非常隐蔽，传统模型难以从高维输入中有效抽取关键特征时，需要人的经验或技术手段辅助抽取重要特征，特征工程模块可起到关键作用。由于特征工程属于机器学习的外围模块，与各种领域知识有密切关系，不作为本书的重点，本小节仅给出一个极为简略的介绍。

特征工程可分为特征选择和特征提取。特征选择指从已有输入向量中，选择一部分特征保

留，删除其他特征分量。特征提取指对原输入向量进行变换，变换后得到降维的特征向量表示。一般特征选择是通过丢弃一些输入分量得以降低维度，而特征提取是将源分量进行了组合，并没有直接删除输入分量。

1. 特征选择

进行特征选择的可能原因有几条[①]：

（1）提高模型预测的准确性；

（2）去除不相关的分量；

（3）提高学习效率，减少计算和存储需求；

（4）降低以后数据收集的成本，只测量与输出有关的变量；

（5）降低模型复杂性，提供改进的可理解的数据和模型。

可定义输入向量中一个分量（一个特征）的关联和冗余，这里，关联表示一个特征与模型输出的关系，冗余表示特征分量之间的关系。一个特征可分为“强关联”“弱关联”“无关联”和“冗余”。一个特征 X 称为“强关联”指模型输出紧密依赖该特征，不能删除该特征；“弱关联”指该特征并不总是必要的，但它对某些子集来说是必要的；“无关联”指其与输出和预测是无关的，删去不影响模型性能；“冗余”是指输入向量的分量之间存在高度相关。一般可保留“强关联”和“弱关联”特征，删除“无关联”特征，在互为相关的冗余特征中至少保留 1 个。

实现特征选择的方法主要有三类：过滤法、包装法和嵌入法。

过滤法：这种方法用变量的统计特性过滤信息量小的变量，该方法与模型的学习过程无关，仅依赖于对训练数据一般特性的测度，如距离和相关性等。

过滤法的基本思想是利用距离、相关或概率等度量，筛选最有信息量的分量，有多种实现的具体算法，这里给出两个简单例子说明。第一个例子为设输入向量中的两个分量为 X 和 Y，将这两个分量在训练集中的所有取值构成集合 $\{x_n\}_{n=1}^N$ 和 $\{y_n\}_{n=1}^N$（注意，这两个集合相当于式(4.1.12)中数据矩阵的两列），其均值分别为 $\bar{x}$ 和 $\bar{y}$，则可定义相关系数为

$$\rho(X,Y)=\frac{\sum_{n=1}^{N}(x_n-\bar{x})(y_n-\bar{y})}{\left[\sum_{n=1}^{N}(x_n-\bar{x})^2\sum_{n=1}^{N}(y_n-\bar{y})^2\right]^{1/2}} \tag{5.1.4}$$

其中，$-1\leqslant\rho(X,Y)\leqslant 1$，若 $\rho(X,Y)=\pm 1$，则 X,Y 完全相关，则可删除其中一个分量。

另一个简单例子是，将每个输入分量与输出标注计算相关，并按相关系数绝对值排序，选择排在前面的分量。

实际中，人们提出众多的距离函数，以分散性好为目标选择特征子集。

包装法：用模型的效果评价特征子集，用模型的预测准确性来衡量特征子集的优劣。该方法选择部分特征分量作为一个子集，用于训练模型，并用独立的测试集来评价模型性能，因此复杂性高很多，但比过滤法选择的特征子集更有效。

有几种选择特征子集的方法。性能最好的方法是穷举法，即将所有子集的组合依次用于训练模型，然后用测试集测试性能，找到性能最好的特征子集。在实际中，可以用分支界定法实现全搜索。次优的方法是顺序法，即按顺序依次增加或删除特征（对应连续前向选择或连续后向选择），比穷举法效率高。还有各种设计的随机搜索方法。

---

① 见本书参考文献[168]。

嵌入法：这种特征选择的方法是嵌入模型训练过程中的，是一种模型相关方法。可以视为在特征子集和模型结构形成的组合空间中搜索，一种典型的方法为决策树（第 7 章）。

2. 特征提取

特征提取的最重要方法之一是主分量分析（PCA），第 13 章将详细介绍其算法。PCA 是典型的无监督学习算法，将高维向量映射为一个低维向量的同时，最大可能保持向量的能量。对于很多高维向量，通过 PCA 降维后作为模型输入的特征向量，可在有效降低模型复杂度和训练复杂度的情况下，不会明显降低模型性能。PCA 是高维分量的一种有效降维表示，在深度学习广泛应用之前，PCA 结合传统模型构成了表示学习的一种模式。

第 4 章介绍的 Fisher 线性判别分析本质上是另一种降维方法，当分类问题共有 C 种类型时，其将输入向量降维为 $C-1$ 维向量。以各类样本子集的最大可分离度为原则，将原样本集映射到 $C-1$ 维空间，由于输入特征向量降维为 $C-1$ 维，而标注不变，为在低维情况下设计分类器提供了一种模式。

在信号处理中大量使用的各种变换技术，也为特征提取提供了技术基础。一些高维向量在原表示域中存在高冗余，当选择合适的变换技术，将其变换到变换域中，其在变换域是稀疏的或近似稀疏的，只需保留部分变换系数组成低维向量即可逼近原向量。常用的变换有离散傅里叶变换（DFT）、KL 变换和 DCT 变换等。如果希望利用特定的结构信息，如多分辨结构，则可采用离散小波变换（DWT）等。关于各类变换的详细讨论，可参考相关的信号处理著作。

在不同的应用领域，存在许多基于领域知识构造的各类特征，这些特征基于对高维原输入向量的各种全局或局部计算，包括线性或非线性运算。

在模式识别和统计学中，特征工程问题被广泛研究。限于篇幅和本书的主题，这里只对特征工程做了非常简略的介绍。

5.1.5 样本不平衡

在进行模型训练之前对样本不平衡问题进行处理，可看作预处理的组成部分，因其重要性单独讨论。

在一些专业问题的有监督学习数据集中，常存在样本的不平衡问题。例如，在检查某一疑难病症的数据集中，标有“患病”的样本（称为正样）数目远少于“非患病”的样本（称为负样）数目，这是样本不平衡的情况。在样本不平衡情况下，很多模型训练的结果是对负样的性能更好，对正样的性能更差，这与需求是非常不一致的。

毫无疑问，解决样本不平衡的根本办法是争取采集更多样本，使样本集平衡，但当这种努力暂时无法达到时，一些辅助技术可适度缓解该问题。

一种最直接的方法是对原样本集进行扩大，将每个数量少的类型的样本重复多次复制到样本集随机的位置。例如，一个样本集正样和负样的比例是 1∶5，将每个正样复制 3 次到样本集的随机位置，则比例变成 4∶5。尽管这种复制没有增加实质正样数量，但结合一些训练算法可有效提升正样的作用。例如，在第 4 章介绍的小批量 SGD 算法中（深度学习目前大多采用该算法做优化），每次随机的从样本集采集一个小批量样本，用这种复制后的数据集，随机采样的小批量样本集中正样的比例明显扩大，正样的作用得到提升。第 8 章介绍的随机森林算法中，每次从样本集中采样一个自助样本集训练一颗决策树，这种加了复制样本集的方法可提升正样的作用。

一个相反的思路是对多的样本进行抽取。如前例所述，负样多出数倍，则可随机抽取出部分负样并将其删除掉。但这种随机抽取不一定有明显效果，可采用可视化处理，检查正样和负样的

几何分布，若几何分布中有较多不同标准的样本重合，可优先将几何位置重叠区域上的负样删除，效果可能更明显。

当样本集的特征向量做少量调整不影响标注的正确性时，可通过微调特征向量部分，获得增广样本。例如，对于图像样本，做少量平移和旋转运算，不影响标注的正确性。在特征向量是来自传感器采集的数据时，加入少量噪声一般不影响标注。在可能的环境下，通过增广样本进行样本平衡，往往有实质性收益。但有些领域，这种调整是不被允许的，例如医学诊断数据。

对于一些模型的训练算法，可将样本加权，这种情况下，可通过对样本加权改善样本不平衡的负面效果。

解决样本不平衡是一个困难的问题，除了在预处理阶段的工作，可进一步结合模型的目标函数、模型结构和训练算法加以改善。

5.2　机器学习模型的性能评估

一个机器学习模型确定后，性能是否符合任务的需求，需要对其进行评估。一般来讲，对于较复杂的实际任务，性能评估方法可能与任务是相关的，因此有关性能评估方式有很多，本书作为以机器学习算法为主的基本教材，不对各种与任务相关的评价方法做过多讨论，本小节只对几个最基本的性能评价方法做一概要介绍，并只讨论监督学习中的回归和分类的性能评估。

对一个机器学习模型 $h(\boldsymbol{x})$ 做准确的性能评估是困难的，实际中一般是在样本集（例如测试集）

$$\boldsymbol{D}=\{(\boldsymbol{x}_i,y_i)\}_{i=1}^{N} \tag{5.2.1}$$

上对其进行性能评估，当样本集中样本数充分多且可充分表示实际样本分布时，用在样本集上的评估作为近似的泛化性能评估。本节为了叙述简单，假设样本集中的标注 y 均是标量。

1. 回归的性能评估

对于回归问题，模型 $h(\boldsymbol{x})$ 的输出和样本标注均为实数，在样本集上评价其性能的常用方法是均方误差，即

$$E_{\text{mse}}(h)=\frac{1}{N}\sum_{i=1}^{N}(h(\boldsymbol{x}_i)-y_i)^2 \tag{5.2.2}$$

均方误差重点关注了大误差的影响，在一些应用中，也可能采用平均绝对误差或最大误差，分别表示为

$$E_{\text{abs}}(h)=\frac{1}{N}\sum_{i=1}^{N}|h(\boldsymbol{x}_i)-y_i| \tag{5.2.3}$$

$$E_{\infty}(h)=\max_{1\leqslant i\leqslant N}\{|h(\boldsymbol{x}_i)-y_i|\} \tag{5.2.4}$$

尽管存在一些其他评价函数，在回归问题中，以均方误差评价使用最多。

2. 分类的性能评估

分类的性能评估比回归要复杂，这里讨论只有两类的情况，在二分类问题中，可用 1、0 分别标注两种不同类型，也常用正类（正样）或负类（负样）表示两种类型。一般正类指在“是”与“否”的二分类中确认为“是”的类型，可用标注 1 表示。例如在判断是否为某种疾病的分类系统中，一般将患有该疾病的样本称为正类，没患该疾病的样本称为负类。

评价分类的最基本准则是分类错误率和分类准确率，当对一个样本 $(\boldsymbol{x}_i,y_i)$ 做分类测试时，若分类器输出 $h(\boldsymbol{x}_i)$ 与样本标注相等，则分类正确，否则产生一个分类错误。对于式(5.2.1)的样本集，统计对所有样本 $h(\boldsymbol{x})$ 能够进行正确分类和错误分类的比例，可得到在样本集上的分类错误率

和分类准确率，分别表示为

$$E=\frac{1}{N}\sum_{i=1}^{N}I(h(\boldsymbol{x}_i)\neq y_i) \tag{5.2.5}$$

$$\mathrm{Acc}=\frac{1}{N}\sum_{i=1}^{N}I(h(\boldsymbol{x}_i)=y_i)=1-E \tag{5.2.6}$$

其中，$I(x)$是示性函数，x 是逻辑变量，当 x 为真 $I(x)=1$，否则 $I(x)=0$。若一个样本集中正样和负样分布均衡(大致数目相当)，各种分类错误(将正样错分为负类或反之)代价相当，分类准确率(或分类错误率)可较好地评价分类器的性能。但当式(5.2.1)所示的样本集中正样和负样分布很不均衡时，分类正确率不能客观地反映分类器性能，甚至会引起误导。例如，在一个检测某癌症的数据集中有一万个样本，正样(癌症患者)的数目只有 300，其余的均为负样(非癌症患者)，对于这样的样本集，若一个分类器简单地将所有样本分类为负样，则分类准确率仍为 0.97，这个指标相当好，但对本任务该分类器毫无用处。

为了进一步讨论怎样构造更合理的评价方法，对于一个分类器 $h(\boldsymbol{x})$，将式(5.2.1)的样本集分为 4 类：(1)“真正类”，样本为正样，分类器将其分为正类；(2)“真负类”，样本是负样，分类器将其分类为负类；(3)“假负类”，样本为正样，分类器将其分为负类；(4)“假正类”，样本是负样，分类器将其分类为正类。样本集中各类的数目见表 5.2.1。

表 5.2.1 样本的类型

标注的真实类型	分类器返回的类型	
	正类	负类
正类	N_{TP}	N_{FN}
负类	N_{FP}	N_{TN}

用表 5.2.1 的符号，样本总数 $N=N_{\mathrm{FP}}+N_{\mathrm{FN}}+N_{\mathrm{TP}}+N_{\mathrm{TN}}$，可重写分类错误率和分类准确率为

$$E=\frac{N_{\mathrm{FP}}+N_{\mathrm{FN}}}{N_{\mathrm{FP}}+N_{\mathrm{FN}}+N_{\mathrm{TP}}+N_{\mathrm{TN}}}$$

$$\mathrm{Acc}=\frac{N_{\mathrm{TP}}+N_{\mathrm{TN}}}{N_{\mathrm{FP}}+N_{\mathrm{FN}}+N_{\mathrm{TP}}+N_{\mathrm{TN}}}$$

对于前述癌症的例子，若将所有样本均分类为负样，则 $N_{\mathrm{FP}}=N_{\mathrm{TP}}=0$，$N_{\mathrm{FN}}=300$，$N_{\mathrm{TN}}=9700$，$E=0.03$，$\mathrm{Acc}=0.97$。在这种情况下，分类错误率和分类准确率几乎无法告诉我们分类器的实际效用，如下定义两个更有针对性的性能评价：精度(Precision)和查全率(Recall)。

精度定义为真正类 N_{TP} 与被分类器识别为正类的所有样本 $N_{\mathrm{TP}}+N_{\mathrm{FP}}$ 的比例，即

$$\mathrm{Pr}=\frac{N_{\mathrm{TP}}}{N_{\mathrm{TP}}+N_{\mathrm{FP}}} \tag{5.2.7}$$

查全率定义为正样被分类器正确识别为正类的概率，即真正类数目 N_{TP} 与正样总数 $N_{\mathrm{TP}}+N_{\mathrm{FN}}$ 之比

$$\mathrm{Re}=\frac{N_{\mathrm{TP}}}{N_{\mathrm{TP}}+N_{\mathrm{FN}}} \tag{5.2.8}$$

首先通过一个例子说明两个参数的意义如下。

例 5.2.1 针对癌症的例子，设一个分类器对样本集的分类情况如表 5.2.2“/”左侧所示。

表 5.2.2　样本的类型

标注的真实类型	分类器返回的类型	
	正类	负类
正类	210/260	90/40
负类	200/400	9500/9300

可计算出精度和查全率分别为 Pr≈0.51 和 Re=0.7，分类准确率 Acc=0.971。

若分类器改变参数，使得输出为正类的概率提高，则可能同时负样被判定为正类的数目也增加了，如表中“/”右侧的数据，则精度和查全率分别为 Pr≈0.39 和 Re≈0.87，分类准确率 Acc=0.956。

例 5.2.1 给出了两组数据，所代表的分类器均比将所有样本都分类为负类的“负分类器”有价值，但就分类准确率来讲，第一组数据没有改善，第二组数据反而下降了。对两组数据自身做比较可见，第二组（表 5.2.2 中“/”右侧）数据将更多的正样（癌症患者）做了正确分类，但同时也将更多的负样判别为正类，因此查全率增加但精度降低了。对于该任务来讲，可以认为第二组数据表示的分类器更有用，它将更多的患者检查出来，以免耽误治疗，对于将负样判别为正类的错误分类，一般可通过后续检查予以改正。在这个应用任务中查全率的提高是更有意义的，但也有的任务希望有更高的精度。在实际中精度和查全率往往是矛盾的，哪一个指标更重要往往取决于具体任务的需求。

可以将精度和查全率综合在一个公式中，即如下的 F_β

$$F_\beta=\frac{(\beta^2+1)\times \mathrm{Pr}\times \mathrm{Re}}{\beta^2\times \mathrm{Pr}+\mathrm{Re}} \tag{5.2.9}$$

对于 F_β，当 $\beta>1$ 时，查全率将得到更大权重，当 $0\leqslant\beta<1$ 时，精度得到更大权重。当 $\beta=1$ 时查全率和精度有相同的权重，得到一个简单的综合性能指标

$$F_1=\frac{2\times \mathrm{Pr}\times \mathrm{Re}}{\mathrm{Pr}+\mathrm{Re}} \tag{5.2.10}$$

当调整一个分类器的参数使得其性能变化时，常利用 P-R 曲线或接收机工作特性（Receiver Operating Characteristic，ROC）曲线评价分类器在不同参数下的表现。

P-R 曲线是以精度为纵轴、查全率为横轴的曲线，一般随着查全率增加，精度下降，图 5.2.1 所示为一个典型 P-R 曲线的示意图。

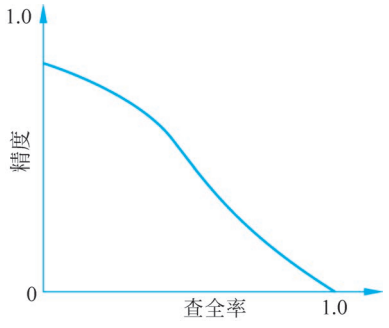


图 5.2.1　P-R 曲线示意

ROC 最初来自雷达检测技术，对于分类器，可首先定义正样分类准确率

$$\mathrm{P_{Ac}}=\frac{N_{\mathrm{TP}}}{N_{\mathrm{TP}}+N_{\mathrm{FN}}} \tag{5.2.11}$$

和负样错误率

$$N_{\mathrm{e}}=\frac{N_{\mathrm{FP}}}{N_{\mathrm{TN}}+N_{\mathrm{FP}}} \tag{5.2.12}$$

当改变分类器参数时，$\mathrm{P_{Ac}}$ 和 N_{e} 都变化，以 $\mathrm{P_{Ac}}$ 为纵轴，以 N_{e} 为横轴，可画出一条曲线，称为 ROC 曲线。一个理想的分类器，可取到 $\mathrm{P_{Ac}}=1$ 和 $N_{\mathrm{e}}=0$ 的点，但在现实中难以实现这样的分类器。实际分类器曲线示例如图 5.2.2 所示[①]。

① 见本书参考文献[86]。

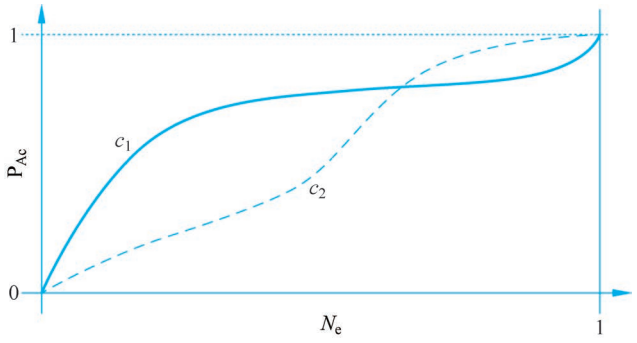


图 5.2.2　分类器 ROC 曲线示例

对于一个实际分类器，若控制参数将所有样本分类为负类，则对应 $(N_e, P_{Ac}) = (0,0)$，若将所有样本分类为正样，则 $(N_e, P_{Ac}) = (1,1)$，则曲线可通过坐标原点和(1,1)点。设有两个分类器(分类器 1 和分类器 2)对应的 ROC 曲线 c_1 和 c_2 画在同一图中，若 c_1 总是位于 c_2 之上，则分类器 1 性能总是优于分类器 2，若两条曲线有交叉，则在不同参数下两个分类器表现各有优劣。对于 ROC 曲线有交叉的不同分类器，一种比较其总体优劣的方法是采用 ROC 曲线下面积(Area Under ROC Curve，AUC)参数，一个分类器的 AUC 参数表示为其 ROC 曲线之下和坐标横轴之间的面积。

针对各类应用任务还有许多特别的参数，例如对于医学应用和对于金融应用，人们关注的性能可能非常不同，本书不再进一步讨论针对实际任务的性能评价。

5.3　机器学习模型的误差分解

本节以回归学习作为对象，讨论模型复杂度与误差的关系，即模型的偏和方差的折中问题。在第 4.3 节的多项式基函数例子中(例 4.3.3)，我们已经看到对于固定规模的训练数据集，随着模型复杂度变化，训练误差和测试误差的变化关系，为了清楚起见，将图 4.3.1(d)重示于图 5.3.1(a)中。在例子中，以多项式阶 M 表示模型的复杂度，可以看到，随着 M 增加，训练误差单调减少，但测试误差先减小，然后上升，也就是模型出现了过拟合。用测试误差逼近所学模型的泛化误差，故模型的泛化误差不是随着模型的表达能力越强而越小，一般泛化误差与模型复杂度的关系是一个“U”形曲线。图 5.3.1(b)所示为训练误差与测试误差的一种更一般的示意图，横坐标表示模型的复杂性度，类似地，训练误差单调减，测试误差是“U”形曲线。对于一个给定的学习任务，可选择对应测试误差“U”形曲线底端的模型复杂度。对于具体例子而言，图 5.3.1(a)中 M 取 3～7 这样一个较宽的范围，测试误差都处于“U”形曲线的底端，都是可选的模型复杂度。

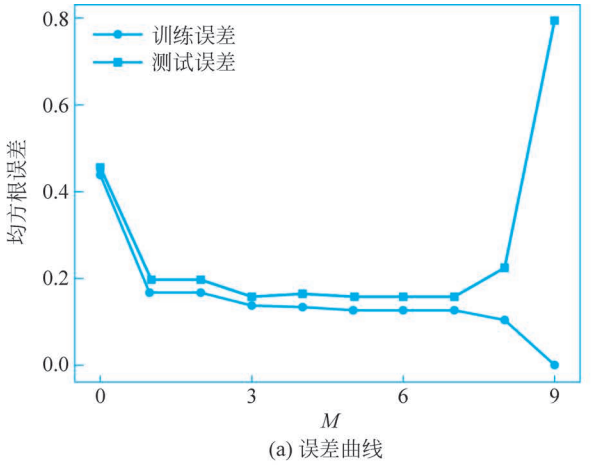
接下来讨论更一般的泛化误差问题，以一般的回归模型作为讨论对象，不限于本书前面讨论的线性回归或基函数回归。假设一个数据集 $\boldsymbol{D} = \{(\boldsymbol{x}_n, y_n)\}_{n=1}^{N}$ 是来自对一个联合概率密度函数 $p(\boldsymbol{x}, y)$ 的采样，通过一个数据集学习得到的模型是 $\hat{y}(\boldsymbol{x})$，注意这里模型没有显式地依赖参数 $\boldsymbol{w}$，可表示更一般的模型。定义误差函数为

$$L(\hat{y}(\boldsymbol{x}), y) = (\hat{y}(\boldsymbol{x}) - y)^2 \tag{5.3.1}$$

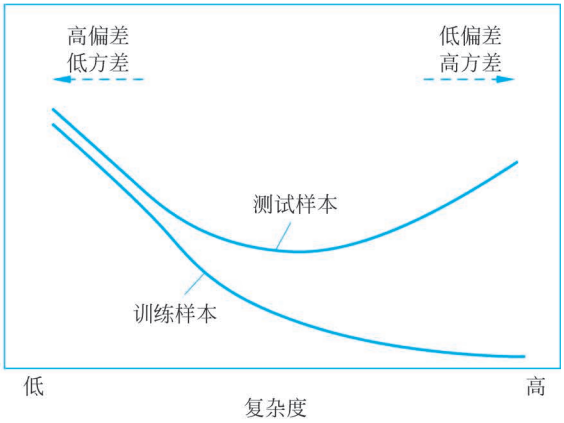
其中，y 表示回归模型要逼近的真实值，针对 $p(\boldsymbol{x}, y)$ 可得到模型的误差期望为

$$E(L) = \iint L(\hat{y}(\boldsymbol{x}), y) p(\boldsymbol{x}, y) \mathrm{d}\boldsymbol{x} \mathrm{d}y = \iint (\hat{y}(\boldsymbol{x}) - y)^2 p(\boldsymbol{x}, y) \mathrm{d}\boldsymbol{x} \mathrm{d}y \tag{5.3.2}$$

其中，$E(L)$ 是针对模型 $\hat{y}(\boldsymbol{x})$ 的泛化误差。



(a) 误差曲线



(b) 更一般的误差曲线

图 5.3.1　模型复杂度与误差的关系

对于回归问题，由第 3 章讨论的决策理论可知，若已知 $p(\boldsymbol{x},y)$，则最优的回归模型为

$$h(\boldsymbol{x})=\int yp(y\mid\boldsymbol{x})\,\mathrm{d}y=E(y\mid\boldsymbol{x}) \tag{5.3.3}$$

在机器学习中，由于一般准确的 $p(y\mid\boldsymbol{x})$ 是未知的，无法直接获得最优回归模型 $h(\boldsymbol{x})$，但是从原理上可以将 $h(\boldsymbol{x})$ 作为一个比较基准，得到如下误差分解：

$$\begin{aligned}E(L)&=\iint(\hat{y}(\boldsymbol{x})-y)^2p(\boldsymbol{x},y)\mathrm{d}\boldsymbol{x}\,\mathrm{d}y\\&=\iint(\hat{y}(\boldsymbol{x})-h(\boldsymbol{x})+h(\boldsymbol{x})-y)^2p(\boldsymbol{x},y)\mathrm{d}\boldsymbol{x}\,\mathrm{d}y\\&=\iint(\hat{y}(\boldsymbol{x})-h(\boldsymbol{x}))^2p(\boldsymbol{x},y)\mathrm{d}\boldsymbol{x}\,\mathrm{d}y+\iint(h(\boldsymbol{x})-y)^2p(\boldsymbol{x},y)\mathrm{d}\boldsymbol{x}\,\mathrm{d}y+\\&\quad 2\iint(\hat{y}(\boldsymbol{x})-h(\boldsymbol{x}))(h(\boldsymbol{x})-y)p(\boldsymbol{x},y)\mathrm{d}\boldsymbol{x}\,\mathrm{d}y\\&=\int(\hat{y}(\boldsymbol{x})-h(\boldsymbol{x}))^2p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}+\iint(E(y\mid\boldsymbol{x})-y)^2p(\boldsymbol{x},y)\mathrm{d}\boldsymbol{x}\,\mathrm{d}y\end{aligned} \tag{5.3.4}$$

其中，$\hat{y}(\boldsymbol{x})-h(\boldsymbol{x})$ 与 y 无关，交叉项积分为 0，误差项由两项组成。其中，第二项是随机变量的不可完全预测性的结果，这是一个固有量，与模型选择、学习过程均无关，式(5.3.4)中最后一行的第一项是与模型和学习过程有关的，接下来仔细分析这一项。

假设只给出一个数据集 $\boldsymbol{D}=\{(\boldsymbol{x}_n,y_n)\}_{n=1}^N$，在这个数据集上学习回归模型，学习到的模型是与数据集相关的，为了清楚表示其与数据集的相关性，将学习到的模型表示为 $\hat{y}(\boldsymbol{x};\boldsymbol{D})$，若可以获得若干数据集，将每个数据集学习的模型做平均，当数据集数目很大时，这个平均值逼近于一个期望，用符号 $E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))$ 表示。使用这个符号，将针对一个指定数据集的误差项 $(\hat{y}(\boldsymbol{x})-h(\boldsymbol{x}))^2$ 分解为

$$\begin{aligned}(\hat{y}(\boldsymbol{x};\boldsymbol{D})-h(\boldsymbol{x}))^2&=(\hat{y}(\boldsymbol{x};\boldsymbol{D})-E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))+E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))-h(\boldsymbol{x}))^2\\&=[\hat{y}(\boldsymbol{x};\boldsymbol{D})-E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))]^2+[E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))-h(\boldsymbol{x})]^2+\\&\quad 2[\hat{y}(\boldsymbol{x};\boldsymbol{D})-E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))][E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))-h(\boldsymbol{x})]\end{aligned}\tag{5.3.5}$$

将以上误差项对所有不同数据集进行平均，即取 $E_D(\cdot)$，注意到交叉项为 0，故

$$\begin{aligned}E_D[(\hat{y}(\boldsymbol{x};\boldsymbol{D})-h(\boldsymbol{x}))^2]&=E_D\{[\hat{y}(\boldsymbol{x};\boldsymbol{D})-E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))]^2\}+E_D\{[E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))-h(\boldsymbol{x})]^2\}\\&=[E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))-h(\boldsymbol{x})]^2+E_D\{[\hat{y}(\boldsymbol{x};\boldsymbol{D})-E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))]^2\}\end{aligned}\tag{5.3.6}$$

式(5.3.6)第二行的第一项是偏差，从多个数据集分别学习得到模型的 $\hat{y}(\boldsymbol{x};\boldsymbol{D})$ 做平均的结果 $E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))$ 仍与最优模型 $h(\boldsymbol{x})$ 之间存在偏差；第二项是学习得到的模型的方差，即每一个数据集训练得到的模型与模型期望之间的偏离程度，这个方差越大，不同数据集训练出的模型的起伏程度越大。由于式(5.3.6)对所有数据集 $\boldsymbol{D}$ 取了期望，其与具体数据集无关，可以看作 $(\hat{y}(\boldsymbol{x})-h(\boldsymbol{x}))^2$ 并代入式(5.3.4)得到

$$\begin{aligned}E(L)&=\int[E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))-h(\boldsymbol{x})]^2p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}+\int E_D\{[\hat{y}(\boldsymbol{x};\boldsymbol{D})-E_D(\hat{y}(\boldsymbol{x};\boldsymbol{D}))]^2\}p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}+\\&\quad\iint(E(y\mid\boldsymbol{x})-y)^2p(\boldsymbol{x},y)\mathrm{d}\boldsymbol{x}\mathrm{d}y\\&=(\text{偏差})^2+\text{方差}+\text{固有误差}\end{aligned}\tag{5.3.7}$$

式(5.3.7)说明，对于一个给出的模型，其泛化误差由三部分组成，偏差(实际是偏差的平方，为了叙述简单这里称为偏差)、方差和固有误差。固有误差与模型、数据集和学习过程均无关，不需要进一步讨论。偏差和方差确实与模型选择有关，一般的，若选择比较简单的模型，则偏差比较大，这是由于模型的表示能力有限，即使从多次训练获得的模型平均也仍偏离最优模型 $h(\boldsymbol{x})$。若选择比较复杂的模型，可以使得偏差比较小，但方差变大。设模型是参数模型，则复杂模型具有更多的参数，在给定数据集规模的条件下，每个参数的平均有效样本数较小。第 2 章讨论过，方差与有效样本数成反比，因此方差变大。即模型简单，方差小，偏差大；模型复杂，方差大，偏差小。当模型取得比较合适，既不算复杂也不算简单，即相对折中的模型，可能偏差和方差都比较小，总误差最小。图 5.3.2 所示为偏差、方差和泛化误差随模型复杂度的变化曲线的示意图。

例 5.3.1 为了给出误差分解的直观理解，考虑一个简单的学习模型的例子。

设函数 $f(\boldsymbol{x})$ 是无法直接观测到的，为了对该函数进行预测，通过采样获得数据集，采样过程为

$$y=f(\boldsymbol{x})+v\tag{5.3.8}$$

由于无法直接观测 $f(\boldsymbol{x})$，故采样样本存在误差 v，设 v 为零均值、方差为 σ_v^2 的高斯噪声。为了讨论问题简单，采样时各输入 $\boldsymbol{x}_i$ 是预先确定的。由采样数据构成 IID 数据集 $\{\boldsymbol{x}_i,y_i\}_{i=1}^N$，由数据集训练一个模型，作为说明，这里采用 K 近邻回归算法，模型为

$$\hat{y}=\hat{f}(\boldsymbol{x})=\frac{1}{K}\sum_{l=1}^{K}y_{(l)}\tag{5.3.9}$$

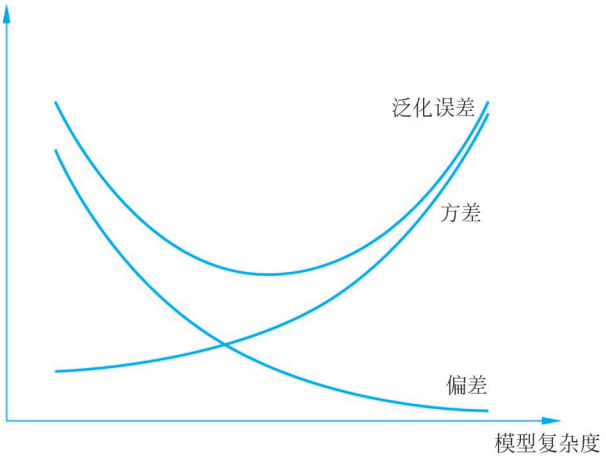


图 5.3.2　模型的误差分解

其中，$y_{(l)}$ 表示对于给定的 $\boldsymbol{x}$，最近邻的 K 个训练集样本的标注值，(l) 表示最近邻样本的下标。

为了讨论误差分解，通过对式(5.3.8)的观测，可得到最优模型为

$$h(\boldsymbol{x})=E(y\mid\boldsymbol{x})=f(\boldsymbol{x})$$

因此，固有误差为 σ_v^2。由于本例较为简单，直接使用式(5.3.4)的最后一行，则有

$$\begin{aligned}E(L)&=E\{(\hat{y}-h(\boldsymbol{x}))^2\}+\sigma_v^2\\&=E\left\{\left(\frac{1}{K}\left(\sum_{l=1}^{K}f(\boldsymbol{x}_{(l)})+v_l\right)-f(\boldsymbol{x})\right)^2\right\}+\sigma_v^2\\&=E\left\{\left[\frac{1}{K}\sum_{l=1}^{K}f(\boldsymbol{x}_{(l)})-f(\boldsymbol{x})\right]^2\right\}+E\left\{\left(\frac{1}{K}\sum_{l=1}^{K}v_l\right)^2\right\}+\sigma_v^2\\&=\left[\frac{1}{K}\sum_{l=1}^{K}f(\boldsymbol{x}_{(l)})-f(\boldsymbol{x})\right]^2+\frac{\sigma_v^2}{K}+\sigma_v^2\end{aligned}\tag{5.3.10}$$

式(5.3.10)从倒数第二行到最后一行，是考虑做预测时，$\boldsymbol{x}$ 是一个给出的固定值。式(5.3.10)的最后一行分别是如式(5.3.7)所示的：偏差、方差和固有误差。对于 K 近邻方法，K 越大代表模型表达力越弱，$K=1$ 表示表达能力最强。显然，对于变化的内在函数 $f(\boldsymbol{x})$，K 越大 $\frac{1}{K}\sum_{l=1}^{K}f(\boldsymbol{x}_{(l)})$ 与 $f(\boldsymbol{x})$ 偏差越大(越多偏离 $\boldsymbol{x}$ 更远的函数值参与平均)，但方差 $\frac{\sigma_v^2}{K}$ 越小；反之，K 越小，二者的偏差越小，但方差越大，最小时 $K=1$，则由最近邻的函数 $f(\boldsymbol{x}_{(l)})$ 逼近 $f(\boldsymbol{x})$，此时偏差最小，方差 $\frac{\sigma_v^2}{K}$ 最大。不管 K 取何值，最后一项的固有误差 σ_v^2 不变。

对于线性回归模型，也可导出闭式结果说明误差的分解，只是推导过程更加复杂一些。

对于机器学习的模型选择来讲，在处理给定的问题和数据集时，并不是选择越复杂的模型越好，要选择适中的模型。这是一个基本的原则，在实际中怎样使用这个原则，却不是一个简单的问题。从原理上讲，从最大似然原理过渡到完全的贝叶斯框架下，可以解决模型选择的问题，但对于一般非线性模型，贝叶斯框架下的求解要复杂得多。一个更实际的方法是通过正则化和交叉验证来合理地选择模型。

本节注释 图 5.3.1 中的测试误差和图 5.3.2 中的泛化误差曲线都是“U”形曲线。对于传统的单一机器学习模型,“U”形曲线具有一般性。但在深度学习中,当深度网络复杂度达到一定规模后,测试误差的表现更加复杂,对于集成学习中一些方法,如随机森林和提升算法,测试误差一般也并没有呈现出“U”形,换言之,集成学习更不易出现过拟合问题。机器学习是仍在快速发展中的领域,在发展中,一些传统结论可能被不断补充和修改。

5.4 机器学习模型的泛化性能

5.3 节以回归问题为例,讨论了偏差和方差的折中问题,本节将以分类问题为例,讨论机器学习的另一个理论问题:泛化界。偏差和方差的折中与泛化界是机器学习理论中关注的两个基本问题,都是关于泛化误差的,两者之间也有密切的联系。

在机器学习模型的训练过程中,一般只有一组训练集,算法通过训练误差的最小化学习到一个模型,但我们真正关心的是泛化误差,即对不存在于训练集中的新样本来讲,模型的预测性能如何?因此我们要关心一个基本问题:训练误差和泛化误差之间有多大的差距?机器学习的概率近似正确(Probably Approximately Correct, PAC)理论对这个问题进行了研究。这里对该理论给出一个极为简要的介绍。

本节以二分类问题为例,讨论 PAC 理论的一些基本概念和结论。假设样本可表示为$(\boldsymbol{x},y)$,$\boldsymbol{x}\in\boldsymbol{\mathcal{X}}$是输入特征向量,$\boldsymbol{\mathcal{X}}$表示输入空间,$y\in\{0,1\}$表示类型,$(\boldsymbol{x},y)$满足概率分布$p_{\mathcal{D}}(\boldsymbol{x},y)$,简写为$p_{\mathcal{D}}$,故可用$(\boldsymbol{x},y)\sim p_{\mathcal{D}}$表示样本服从的分布。从$p_{\mathcal{D}}$中采样得到满足独立同分布(IID)的训练样本集

$$\boldsymbol{D}=\{(\boldsymbol{x}_i,y_i)\}_{i=1}^{N} \tag{5.4.1}$$

其中,每个样本$(\boldsymbol{x}_i,y_i)\sim p_{\mathcal{D}}$。

用机器学习理论常用的术语,将一个机器学习模型称为一个假设h,$h(\boldsymbol{x})$输出$\{0,1\}$表示类型,即h完成映射:$h:\boldsymbol{\mathcal{X}}\to\{0,1\}$。在一个机器学习过程中,所有可能选择的假设构成一个假设空间$\boldsymbol{\mathcal{H}}$。对于任意假设$h\in\boldsymbol{\mathcal{H}}$,其在训练样本集上的分类错误率定义为训练误差或经验风险,经验风险表示为

$$\hat{R}(h)=\frac{1}{N}\sum_{i=1}^{N}I(h(\boldsymbol{x}_i)\neq y_i) \tag{5.4.2}$$

其表示了假设h在训练集上的误分类率,$I(\cdot)$是示性函数。可定义一般的泛化误差为

$$R(h)=P_{(\boldsymbol{x},y)\sim p_{\mathcal{D}}}(h(\boldsymbol{x})\neq y) \tag{5.4.3}$$

即泛化误差表示任意$(\boldsymbol{x},y)\sim p_{\mathcal{D}}$的误分类率,不管其是否存在于训练集。注意,用$R$表示泛化误差,用$\hat{R}$表示经验风险。

若不考虑可实现性,从理论上讲,我们希望学习到的假设是从$\boldsymbol{\mathcal{H}}$空间找到使得泛化误差最小的假设,即

$$h^{*}=\underset{h\in\boldsymbol{\mathcal{H}}}{\operatorname{argmin}}R(h) \tag{5.4.4}$$

在实际中,由于无法准确获得$p_{\mathcal{D}}(\boldsymbol{x},y)$,故无法通过泛化误差优化获得最优假设,总是通过经验风险最小化(Empirical Risk Minimization, ERM)得到一个假设,即

$$\hat{h}=\underset{h\in\boldsymbol{\mathcal{H}}}{\operatorname{argmin}}\hat{R}(h) \tag{5.4.5}$$

我们关心的一个理论问题是：对于通过 ERM 得到的假设 $\hat{h}$ 与真正的泛化误差最小的 h^* 之间的泛化误差差距有多大？即 $R(h^*)$ 与 $R(\hat{h})$ 差距有多大？

在继续讨论之前，首先通过例子进一步理解以上概念。

例 5.4.1　一个假设空间的例子。在 4.5 节介绍的感知机中，为了与本节分类输出用 $\{0,1\}$ 表示相符，将分类假设修改为

$$h(\boldsymbol{x})=I(\bar{\boldsymbol{w}}^{\mathrm{T}}\bar{\boldsymbol{x}}\geqslant 0) \tag{5.4.6}$$

其中，$\bar{\boldsymbol{x}}$ 包含了哑元，$\bar{\boldsymbol{w}}$ 包含了偏置系数，是 $K+1$ 维参数向量。式(5.4.6)是一个假设，则假设空间为

$$\mathcal{H}=\{h_{\bar{\boldsymbol{w}}}\mid h_{\bar{\boldsymbol{w}}}(\boldsymbol{x})=I(\bar{\boldsymbol{w}}^{\mathrm{T}}\bar{\boldsymbol{x}}\geqslant 0),\bar{\boldsymbol{w}}\in\mathbf{R}^{K+1}\} \tag{5.4.7}$$

其表示 K 维向量空间中的所有线性分类器集合，其中，$\mathbf{R}^{K+1}$ 是 $K+1$ 维实数集合。由于不同 $\bar{\boldsymbol{w}}$ 构成 $\mathcal{H}$ 的不同成员，故式(5.4.5)可具体化为

$$\hat{h}=\underset{\bar{\boldsymbol{w}}\in\mathbf{R}^{K+1}}{\operatorname{argmin}}\hat{R}(h_{\bar{\boldsymbol{w}}}) \tag{5.4.8}$$

$\hat{h}$ 是 ERM 意义下的最优假设，一般不能通过学习得到 $h_{\bar{\boldsymbol{w}}^*}$。

实际上，感知机的目标函数式(4.5.43)是式(5.4.2)的经验风险的一种逼近，故训练得到的感知机是对 $\hat{h}$ 的一种逼近。

逻辑回归也可做类似理解，同样其目标函数交叉熵也是式(5.4.2)经验风险函数的一种近似。

为了研究 $R(h^*)$ 与 $R(\hat{h})$ 的关系，给出如下引理。

引理 5.4.1　设 $Z_1,Z_2,\cdots,Z_N$ 是 N 个独立同分布的随机变量，均服从伯努利分布，且 $P(Z_i=1)=\mu$，定义样本均值为 $\hat{\mu}=\dfrac{1}{N}\sum_{i=1}^{N}Z_i$，令 $\varepsilon>0$ 为一个固定值，则

$$P(|\mu-\hat{\mu}|>\varepsilon)\leqslant 2\exp(-2\varepsilon^2 N) \tag{5.4.9}$$

引理说明对于独立同分布样本，当 N 充分大，对于给定的 $\varepsilon>0$，概率的均值估计和实际概率值之差大于 ε 的概率是随样本数 N 指数减小的。

接下来，对于 $\mathcal{H}$ 有限的情况，利用引理 5.4.1 导出训练误差和泛化误差的误差界，然后将结论推广到 $\mathcal{H}$ 无限的情况。

5.4.1　假设空间有限时的泛化误差界

首先考虑假设空间 $\mathcal{H}$ 是有限的，即 $\mathcal{H}=\{h_1,h_2,\cdots,h_M\}$，假设空间成员数目 $M=|\mathcal{H}|$ 可能很大，但是有限的。例如 4.7 节介绍的朴素贝叶斯方法，其假设空间是有限的。若每个特征变量取有限值，则第 7 章介绍的决策树假设空间也是有限的。对于例 5.4.1，若 $\bar{\boldsymbol{w}}$ 取值为实数，则假设空间是无限的，但若 $\bar{\boldsymbol{w}}$ 的每个分量是由有限位二进制表示的数值，则其假设空间是有限的(详细讨论见稍后的例 5.4.2)。首先讨论 $\mathcal{H}$ 有限的情况。

可从 $\mathcal{H}$ 选择一个固定的假设 h_k，利用引理 5.4.1 容易得到对于 h_k，其训练误差和泛化误差的关系。为此定义一个随机变量，对于 $(\boldsymbol{x},y)\sim p_{\mathcal{D}}$，定义

$$Z=I(h_k(\boldsymbol{x})\neq y) \tag{5.4.10}$$

即当 h_k 对样本 $(\boldsymbol{x},y)$ 不能正确分类时 $Z=1$，对式(5.4.1)所示的每个样本有 $Z_i=I(h_k(\boldsymbol{x}_i)\neq y_i)$，显然，$h_k$ 的训练误差为

$$\hat{R}(h_k)=\frac{1}{N}\sum_{i=1}^{N}Z_j \tag{5.4.11}$$

对比引理 5.4.1，Z_i 是 IID 的伯努利随机变量，$\hat{R}(h_k)$ 是对 $R(h_k)$ 的样本均值估计，则由式(5.4.9)直接得到

$$P(|R(h_k)-\hat{R}(h_k)|>\varepsilon)\leqslant 2\exp(-2\varepsilon^2 N) \tag{5.4.12}$$

以上是对于一个固定的 h_k，泛化误差和训练误差之差(绝对值)大于 ε 的概率。对于给定 ε，若样本数 N 充分大，则泛化误差与训练误差相差大于 ε 的概率很小。

利用式(5.4.12)可导出一个更一般的结果。为此，定义 $|R(h_k)-\hat{R}(h_k)|>\varepsilon$ 为一个事件 A_k，则有 $P(A_k)\leqslant 2\exp(-2\varepsilon^2 N)$。利用概率性质，至少存在一个 h(表示为 $\exists h$)其 $|R(h)-\hat{R}(h)|>\varepsilon$ 的概率为

$$\begin{aligned}P(\exists h\in\mathcal{H},|R(h)-\hat{R}(h)|>\varepsilon)&=P(A_1\cup A_2\cup\cdots\cup A_K)\\&\leqslant\sum_{k=1}^{|\mathcal{H}|}P(A_k)\\&\leqslant\sum_{k=1}^{|\mathcal{H}|}2\exp(-2\varepsilon^2 N)\\&=2|\mathcal{H}|\exp(-2\varepsilon^2 N)\end{aligned} \tag{5.4.13}$$

由于是概率值，由互补性，式(5.4.13)的等价表示是：

$$P(|R(h)-\hat{R}(h)|\leqslant\varepsilon,\forall h\in\mathcal{H})\geqslant 1-2|\mathcal{H}|\exp(-2\varepsilon^2 N) \tag{5.4.14}$$

在式(5.4.14)中，令

$$\delta=2|\mathcal{H}|\exp(-2\varepsilon^2 N) \tag{5.4.15}$$

式(5.4.14)有丰富的内涵，其中有三个变量：δ,ε,N，以下从几方面讨论式(5.4.14)的含义，并讨论这三个变量的关系。

(1) 将式(5.4.14)重写为

$$P(|R(h)-\hat{R}(h)|\leqslant\varepsilon)\geqslant 1-\delta,\quad \forall h\in\mathcal{H} \tag{5.4.16}$$

对于给定的 ε，所有假设 $\forall h\in\mathcal{H}$ 都以不小于 $1-\delta$ 的概率满足 $|R(h)-\hat{R}(h)|\leqslant\varepsilon$，即泛化误差和训练误差之差不大于界 ε。这里 $1-\delta$ 是一个置信概率，当 N 很大时，δ 很小，以很高的概率满足 $|R(h)-\hat{R}(h)|\leqslant\varepsilon$。

(2) 假设空间成员数目 $|\mathcal{H}|$ 是确定的，若给出 ε 和 δ，则可得到满足以 $1-\delta$ 为概率达到 $|R(h)-\hat{R}(h)|\leqslant\varepsilon$ 所需的样本数目。固定 δ,ε，从式(5.4.15)反解 N 为

$$N=\frac{1}{2\varepsilon^2}\ln\frac{2|\mathcal{H}|}{\delta} \tag{5.4.17}$$

由式(5.4.15)可见，若 N 增大，则 δ 减小，故可将式(5.4.17)看作满足 δ,ε 约束的最小样本数，故对于给定 δ,ε，样本数可取为

$$N\geqslant\frac{1}{2\varepsilon^2}\ln\frac{2|\mathcal{H}|}{\delta} \tag{5.4.18}$$

(3) 在式(5.1.15)中，固定 δ,N，解得 ε 为

$$\varepsilon=\sqrt{\frac{1}{2N}\ln\frac{2|\mathcal{H}|}{\delta}} \tag{5.4.19}$$

这里给出式(5.4.16)的另一种解释，对于给定的 δ，N，误差界满足

$$|R(h)-\hat{R}(h)|\leqslant\sqrt{\frac{1}{2N}\ln\frac{2|\mathcal{H}|}{\delta}} \tag{5.4.20}$$

将以上解释总结为如下定理 5.4.1。

定理 5.4.1　对于假设空间 $\mathcal{H}$，固定 δ，N，则以概率不小于 $1-\delta$，泛化误差与训练误差满足

$$|R(h)-\hat{R}(h)|\leqslant\sqrt{\frac{1}{2N}\ln\frac{2|\mathcal{H}|}{\delta}}$$

或固定 δ，ε，若样本数目取

$$N\geqslant\frac{1}{2\varepsilon^2}\ln\frac{2|\mathcal{H}|}{\delta}$$

则，以概率不小于 $1-\delta$ 满足 $|R(h)-\hat{R}(h)|\leqslant\varepsilon$。

以上结论对于假设空间中的任意假设 $\forall h\in\mathcal{H}$ 均成立。我们更感兴趣的一个问题是，对于以经验风险最小化学习得到的假设 $\hat{h}$ 和理论上泛化误差最小对应的 h^* 之间的泛化误差的比较。由以上结果，若对 $\forall h\in\mathcal{H}$ 有 $|R(h)-\hat{R}(h)|\leqslant\varepsilon$，则

$$R(h)\leqslant\hat{R}(h)+\varepsilon \tag{5.4.21}$$

对 $\hat{h}$，可得到如下不等式：

$$\begin{aligned}R(\hat{h})&\leqslant\hat{R}(\hat{h})+\varepsilon\\&\leqslant\hat{R}(h^*)+\varepsilon\\&\leqslant R(h^*)+2\varepsilon\end{aligned} \tag{5.4.22}$$

式(5.4.22)第一行只是将 $\hat{h}$ 代入式(5.4.21)，第 2 行将 $\hat{R}(\hat{h})\leqslant\hat{R}(h^*)$ 代入其中，这是因为 $\hat{h}$ 是经验误差最小的假设，第 3 行对 h^* 再次使用式(5.4.21)。式(5.4.22)的结论是：ERM 学习得到的假设 $\hat{h}$ 和泛化误差最优的 h^*，二者的泛化误差之差不大于 2ε。将该结论总结为重要的定理 5.4.2。

定理 5.4.2　对于假设空间 $\mathcal{H}$，固定 δ，N，则以概率不小于 $1-\delta$，泛化误差满足如下不等式：

$$R(\hat{h})\leqslant\min_{h\in\mathcal{H}}R(h)+2\sqrt{\frac{1}{2N}\ln\frac{2|\mathcal{H}|}{\delta}} \tag{5.4.23}$$

或固定 δ，ε，若样本数目取

$$N\geqslant\frac{1}{2\varepsilon^2}\ln\frac{2|\mathcal{H}|}{\delta}$$

则，以概率不小于 $1-\delta$ 满足 $R(\hat{h})\leqslant\min_{h\in\mathcal{H}}R(h)+2\varepsilon$。

由定理 5.4.1 和定理 5.4.2 可见，若固定 δ，N，式(5.4.19)计算了一个界 ε，对于任意 $h\in\mathcal{H}$，训练误差和泛化误差之差以概率 $1-\delta$ 不大于 ε，同时 EMR 最小化得到的假设 $\hat{h}$ 的泛化误差与最小泛化误差之间的差距不大于 2ε。

例 5.4.2　讨论例 5.4.1 的假设空间

$$\mathcal{H}=\{h_{\bar{w}}\,|\,h_{\bar{w}}(\boldsymbol{x})=I(\bar{\boldsymbol{w}}^{\mathrm{T}}\bar{\boldsymbol{x}}\geqslant 0),\bar{\boldsymbol{w}}\in\mathbf{R}^{K+1}\}$$

其中，$\bar{\boldsymbol{w}}$ 有 $K+1$ 个系数，若 $\bar{\boldsymbol{w}}$ 的每个分量用字长 L 的二进制码表示(计算机中常取 L 为 16、32 或 64，近期一些研究采用低比特实现神经网络时，甚至取 L 为 8、4 甚至 1)，则表示 $\bar{\boldsymbol{w}}$ 所需的二进制码位数为 $L(K+1)$，故假设空间 $\mathcal{H}$ 共有 $|\mathcal{H}|=2^{L(K+1)}$ 个元素，由式(5.4.18)，可得对于固定 δ，ε，样

本数需满足

$$N \geqslant \frac{1}{2\varepsilon^2}\ln\frac{2|\mathcal{H}|}{\delta}=\frac{1}{2\varepsilon^2}\ln\frac{2\times 2^{L(K+1)}}{\delta}=O\left(\frac{KL}{\varepsilon^2}+\ln\frac{1}{\delta}\right)=O_{\varepsilon,\delta}(K) \tag{5.4.24}$$

或记为

$$N \sim O_{\varepsilon,\delta}(K) \tag{5.4.25}$$

这里用 $O(\cdot)$ 表示量级，$O_{\varepsilon,\delta}(\cdot)$ 表示量级函数，其中，δ，ε 是其参数。式(5.4.25)尽管有比例系数存在，但需要的样本数 N 与假设空间成员数目 K 是呈线性关系的。

*5.4.2 假设空间无限时的泛化误差界

将定理 5.4.2 的结论推广到假设空间$\mathcal{H}$无限的情况。如前所述，例 5.4.1 所示的线性模型或后续章节介绍的支持向量机和神经网络等模型，当参数取实数时，假设空间是无限的，即 $|\mathcal{H}|=\infty$，这种情况下式(5.4.23)变得无意义，需要对其进行推广。这里对假设空间无限的情况，只做一个简明扼要的介绍。

当 $|\mathcal{H}|=\infty$ 时，为了表示假设空间的表示能力(或容量)，给出两个概念——打散(Shatters)和 VC 维(Vapnik-Chervonenkis Dimension)，这里只给出其简单直观性的介绍。

首先给出**打散**的概念，对于一个包含 d 个点的集合 $S=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_d\}$，其中 $\boldsymbol{x}_i\in\mathcal{X}$，称$\mathcal{H}$可打散 S 是指：对点集 S 对应加上一个任意标注集 $\{y_1,y_2,\cdots,y_d\}$，则必存在 $h\in\mathcal{H}$，使得 $h(\boldsymbol{x}_i)=y_i$，$i=1,2,\cdots,d$。

对于一个假设空间$\mathcal{H}$，其 **VC 维**的定义为至少存在一个最大元素数为 d 的点集合 S，$\mathcal{H}$可打散 S，则$\mathcal{H}$的 VC 维为 d，记为 $\mathrm{VC}(\mathcal{H})=d$，其中，$d$ 是最大能被$\mathcal{H}$打散的点集的元素数，对于有 $d+1$ 个元素的点集合，$\mathcal{H}$均不可能打散它。

例 5.4.3 图 5.4.1 所示为二维平面上的 3 个点组成的点集和其对应的各种标注，图中的直线表示判决线，假设空间为二维线性分类器，即

$$\mathcal{H}=\{h(\boldsymbol{x})=\theta_1x_1+\theta_2x_2+\theta_0\,|\,\theta_0,\theta_1,\theta_2\in\mathbf{R}\}$$

图中每条判决线属于$\mathcal{H}$，对这个 3 个元素的点集，$\mathcal{H}$可将其打散。如果是 4 个点，对于标注是异或运算，$\mathcal{H}$不能正确分类，故$\mathcal{H}$不能打散 4 个点的点集，因此，平面线性分类器的 VC 维为 3，即 $\mathrm{VC}(\mathcal{H})=3$。

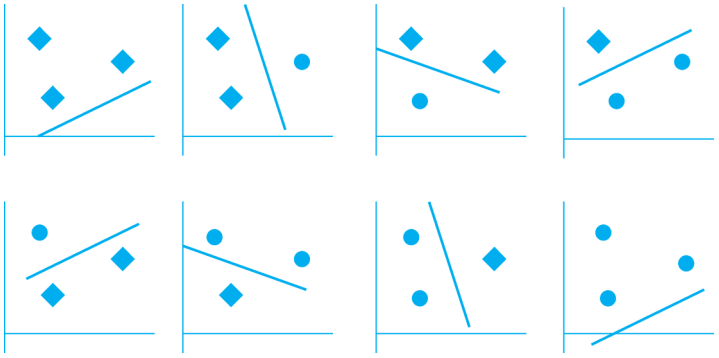


图 5.4.1 可由线性分类器打散的点集

若用 VC 维 d 表示一个假设空间的容量，则可得到与定理 5.4.2 类似的结论，这里只给出对应的定理。

定理 5.4.3　对于假设空间$\mathcal{H}$，若其 VC 维为 $d=\mathrm{VC}(\mathcal{H})$，则对于所有 $h\in\mathcal{H}$，以概率不小于 $1-\delta$，有如下不等式：

$$|R(h)-\hat{R}(h)|\leqslant O\left(\sqrt{\frac{d}{N}\ln\frac{N}{d}+\frac{1}{N}\ln\frac{1}{\delta}}\right) \tag{5.4.26}$$

对于 $\hat{h}$ 有不等式

$$R(\hat{h})\leqslant \min_{h\in\mathcal{H}}R(h)+O\left(\sqrt{\frac{d}{N}\ln\frac{N}{d}+\frac{1}{N}\ln\frac{1}{\delta}}\right) \tag{5.4.27}$$

对于以概率不小于 $1-\delta$ 满足 $R(\hat{h})\leqslant\min_{h\in\mathcal{H}}R(h)+2\varepsilon$，样本数目需满足

$$N\sim O_{\varepsilon,\delta}(d) \tag{5.4.28}$$

注意，这里只用了量级函数 $O(\cdot)$ 而忽视了表达式中的一些具体常数系数，对于了解性能界来讲，这就够了。从式(5.4.28)看，对于无限假设空间，所需样本数与假设空间的 VC 维呈线性关系。

机器学习理论给出对学习性能的整体性洞察是有意义的，但目前对于一类实际模型的学习算法(如逻辑回归、神经网络、决策树等)，其给出的一些要求如样本数等与具体算法的实际需求还有较大距离，在指导一类具体算法的参数选择上的实际意义还有待改善，故实际中仍更常用交叉验证等技术确定机器学习模型的各类参数。

本章小结

本章给出了机器学习的性能与评估的基本介绍，从两方面进行讨论。首先从实用方面，介绍了利用数据集通过交叉验证和测试的实际技术训练一个机器学习模型的基本过程，然后介绍了几个实际中常用的机器学习性能评估指标。然后从理论上讨论了机器学习的性质。为了讨论上的直观，结合回归介绍了偏差和方差的折中问题，结合分类介绍了泛化界定理。

本书更侧重于机器学习算法的介绍，有关机器学习理论的讨论非常简略，对机器学习理论更感兴趣的读者，可参考 Mohri 等的教材和 S. S. Shai 等的著作，这些作品对机器学习理论进行了较为深入的讨论，均由张文生等译成了中文。Vapnik 对于统计学习理论给出了一个简明版的读本，已由张学工译成中文。

本章习题

1. 一个数据集为$\{-3,2.4,1,-2.2,0.8,-1,-1.8,2,4\}$。

(1) 通过取值归一化，将每个输入归一化在$[0,1]$之间；

(2) 通过概率分布归一化，将其归一化。

2. 什么是交叉验证？假设有 10 000 个带标注的样本，设计一个参数模型$\hat{\boldsymbol{y}}=h(\boldsymbol{x};\boldsymbol{w})$，讨论可能选择的样本集划分方式和交叉验证过程。

3. 对于如下表示的线性分类器的假设空间

$$\mathcal{H}=\{h_{\bar{w}}\,|\,h_{\bar{w}}(\boldsymbol{x})=I(\bar{\boldsymbol{w}}^{\mathrm{T}}\bar{\boldsymbol{x}}\geqslant 0),\bar{\boldsymbol{w}}\in\mathbf{R}^{K+1}\}$$

设$\bar{\boldsymbol{w}}$ 有 11 个系数，若$\bar{\boldsymbol{w}}$ 的每个分量用字长 8 比特的二进制码表示，若得到的模型的经验误差和泛化误差的差距以 0.95 的概率不大于 0.05，问样本数至少应取多少？