基　础　篇

第 1 章　绪　　论

> 我们想要的是一台能够从经验中学习的机器（“What we want is a machine that can learn from experience”）。 —— **图灵 (1947)**
>
> 机器学习是一个在非显式编程下赋予计算机学习能力的研究领域（“Machine learning is a field of study that gives computers the ability to learn without being explicitly programmed.”）。 —— **塞缪尔 (1959)**

机器学习技术已经在工程、科学等领域发挥重要作用，成为图像识别、语音识别、自然语言处理、信息推荐等任务的首选方案，也为蛋白质结构预测、药物发现等提供数据分析工具。与此同时，随着数据量的增加以及数据类型的不断丰富，各个领域对机器学习的依赖程度也在逐渐增加，机器学习未来可能会变成像微积分、线性代数一样的基础学科。

本书从概率建模和推断的角度，系统讲述机器学习的基本原理、主要模型和算法，由浅入深，兼顾经典与前沿。本章简要介绍机器学习的概念和分类，并且回答何为概率机器学习、为何选择概率的视角等问题，为后续章节提供铺垫。

1.1　机器学习

1.1.1　什么是机器学习

身处信息时代的今天，人和信息系统紧密相连，产生了大量的数据。例如，位于瑞士的大型强子对撞机每年产生 15PB 的数据[①] [1]；通过各种传感器和智能设备每天收集的数据达到 2.5EB [2]。

然而，数据并不等于知识。为了有效利用数据，需要对其进行有效加工总结出有用的知识和信息，以便用于解决新问题。例如，我们希望从患者的历史医疗病例中发现有价值的规律，并用于新疾病的诊断——对症下药；我们希望从用户浏览互联网的记录中发现用户的上网行为习惯或偏好，并在未来进行有效的信息服务——精

① 1PB 等于 10^{15} 比特；1EB 等于 10^{18} 比特，即 1000PB。

准投放；我们也希望从搜集到的流感病例中预估某地区感染人群的规模和传播的风险——疫情管控。

机器学习的基本目的是让计算机能够从数据中自动提取知识，提升其在预测、决策、知识发现等任务上的性能。机器学习的定义有很多种，这里采用如下定义[3]：

定义 1.1.1 （机器学习） 如果一个计算机程序在任务 T 上的性能 P 随着经验数据 $\mathcal{D}$ 的增加不断提升，我们认为该计算机程序在任务 T 和性能 P 上从经验数据 $\mathcal{D}$ 中进行了学习。

我们把这种能够从经验数据中提升性能的计算机程序称为学习算法。根据 T、P 和 $\mathcal{D}$ 的不同，机器学习包含了多种多样的任务和解决这些任务的算法。下面举两个常见的例子。

例 1.1.1 （掷硬币） 任务 T 是要估计一枚硬币在投掷之后出现正面向上的可能性 μ。为此，可以通过多次抛硬币，观察每次投掷之后的情况，搜集经验数据 $\mathcal{D}$。例如，抛 10 次，观察到 $\mathcal{D} = \{H, H, T, H, T, T, H, T, H, H\}$，其中 H 表示正面向上，T 表示反面向上。通过观察数据 $\mathcal{D}$，可以“学习”得到一个估计值 $\hat{\mu}$——一种直观的估计值是数一下 H 出现的频率，即 $\hat{\mu} = N_H/N$，其中 N_H 表示 H 出现的次数，N 表示投掷的总次数。常见的性能指标 P 为二者之间的误差（例如，绝对误差 $|\hat{\mu} - \mu|$ 或平方误差 $(\hat{\mu} - \mu)^2$）。很显然，当 N 比较小时，估计值 $\hat{\mu}$ 可能很不准确；随着投掷次数的增加，$\hat{\mu}$ 变得越来越准确，也越来越可信。

例 1.1.2 （图像分类） 如图 1.1所示，任务 T 是要学习一个分类器，通过输入图片数据识别熊猫和金丝猴两类动物。为此，需要收集经验数据 $\mathcal{D}$（例如，去动物园拍照或者从互联网上通过授权的方式获取），它应包含多个（假设 N 个）已经标注的熊猫图片和金丝猴图片。这里，我们一般用 $\boldsymbol{x} \in \mathbb{R}^d$ 表示图片对应的 d 维特征向量（如每个像素的灰度值或者人工设计的特征组成的向量），用 y 表示类别标签。在分类任务中，我们一般采用识别精度衡量性能 P。

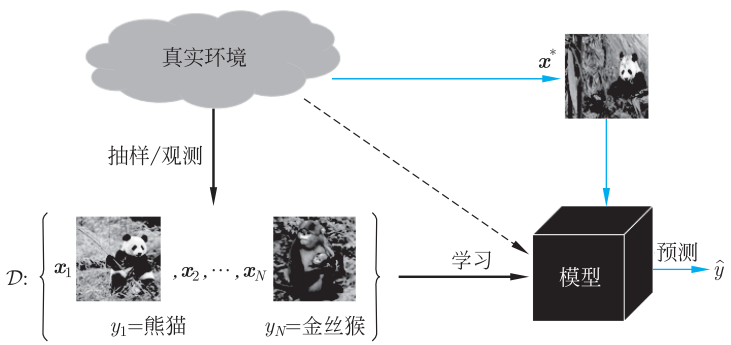


图 1.1 学习一个图像分类器的基本流程，其中，粗黑线表示经验数据获取和模型训练，蓝细线表示模型测试

如果将真实世界中各种可能的熊猫照片和金丝猴照片看作全集，那么，经验数据

$\mathcal{D}$ 实际上是对真实环境（全集）的一个采样（观测）。学习的过程是要建立一个模型，使其尽量“接近”真实世界，以便在给定一个新图片 $\boldsymbol{x}^*$ 时，尽可能准确地识别出是“熊猫”还是“金丝猴”。同样，随着经验数据的增加，在合适的假设条件下，模型往往能学习到真实环境的各种变化，分类精度也将逐渐提升。

我们把学习模型的过程称为“训练”，把使用模型进行预测的过程称为“测试”（也称为推断）。相应地，将训练用的数据集称为训练集，将测试用的数据集称为测试集。

很显然，经验数据 $\mathcal{D}$ 是所有机器学习的一个关键要素，没有数据就谈不上机器学习。但是，只有数据是不够的。

例 1.1.3　(图像分类，续)　假设给定训练数据集 $\mathcal{D}$，如果不加任何限制，一个在训练集上最优的分类器为“死记硬背”——记住每个训练数据的类别，在其他没有看见的数据上总是预测为某一固定类别。很显然，该分类器在新来的测试数据上可能表现非常糟糕！

因此，我们要根据任务 T 从数据中抽取对其有用的信息，并最终完成任务。其中，P 的作用是明确“有用”信息，在这个过程中，我们需要对数据的产生过程、希望获得的预测/决策等做合理的假设（约束），如图 1.1中的虚线所示，这些假设构成了模型。

在上述图像分类的例子中，我们一般假设要寻找的分类器属于某个函数空间 $f \in \mathcal{F}$。该函数空间可以是简单的或复杂的，例如线性回归、深度神经网络，甚至具有无限多个参数的模型等。在给定数据和模型之后，学习算法通过优化某个目标函数，寻找最优拟合数据的模型。对于掷硬币的例子，由于每次投掷的结果具有随机性，因此一个合理的模型是采用概率分布（例如，描述二值变量的伯努利分布）来刻画所观察数据 $\mathcal{D}$ 的可能性，其中分布中的未知参数可以通过数据估计。

1.1.2　机器学习的基本任务

机器学习的任务有很多种，本书将主要介绍 3 种基本任务——有监督学习、无监督学习和强化学习。

有监督学习（supervised learning）

有监督学习的目标是学习一个从数据特征空间 $\mathcal{X}$ 到标签空间 $\mathcal{Y}$ 的映射函数 $f : \mathcal{X} \to \mathcal{Y}$。例如，前面提到的图像分类就是一个有监督学习任务。在这里，训练数据一般表示为

$$\mathcal{D} = \{(\boldsymbol{x}_i, y_i) : \boldsymbol{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}\}_{i=1}^{N}. \tag{1.1}$$

机器学习方法在做一些特定的模型假设之后，依据这些假设可以从数据中学习得到模型。学习到的模型可以对新来的数据 $\boldsymbol{x}^*$ 做预测，即 $y^* = f(\boldsymbol{x}^*)$。

有监督学习的其他例子还包括手写体字符识别、文本分类、语音识别等。如图 1.2所示，手写邮编数字识别将已有的手写邮编数字识别为 0~9 数字中的一个。这个任务的数据是已经分割成单个数字的图片和对应的类别标注，图片中的像素点是数据特征空间，类别标注是数据标签空间。以常用的 MNIST 数据集[①]为例，每张图片的像素点为 $28 \times 28 = 784$ 个，一种简单直接的特征表示为 784 维向量，其中每个元素表示对应像素的灰度值。对于文本数据（如新闻网页），分类的类别可能是“政治”“财经”“娱乐”等。文本数据的一种基本特征表示是词袋（bag-of-words）模型——假设字典中有 V 个单词（也用 V 表示字典），统计每篇文章的每个词出现的个数并组成一个 V 维的非负整数向量，将这个向量作为文档的特征表示。

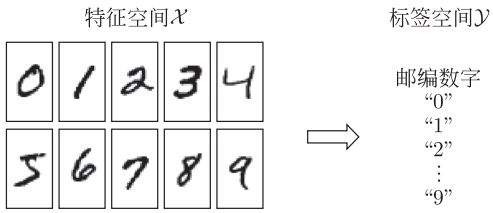


图 1.2　有监督学习的例子：手写邮编数字识别

例 1.1.4　(词袋模型)　假设字典 V 由“我”“爱”“祖”“国”“家”“庭”“幸”“福”“的”共 9 个字组成，则短文本“我爱我家，我爱我的祖国”对应的向量表示为 $\boldsymbol{x} = (4, 2, 1, 1, 1, 0, 0, 0, 1)^\top$, 其中每一个元素表示某个字出现的次数。

由此可见，词袋模型忽略词语的顺序信息，仅仅考虑词语出现的次数。

在有监督学习中，根据标签空间的类型，词袋模型又可细分为分类和回归。具体地，当标签是离散的时候（例如，是/不是垃圾邮件、0~9 数字中的一个），这种任务为分类；当标签是连续的时候（如气温、股票价格等），则为回归任务。对于同一组数据，根据实际需要，可能是分类或回归问题，如例 1.1.5。

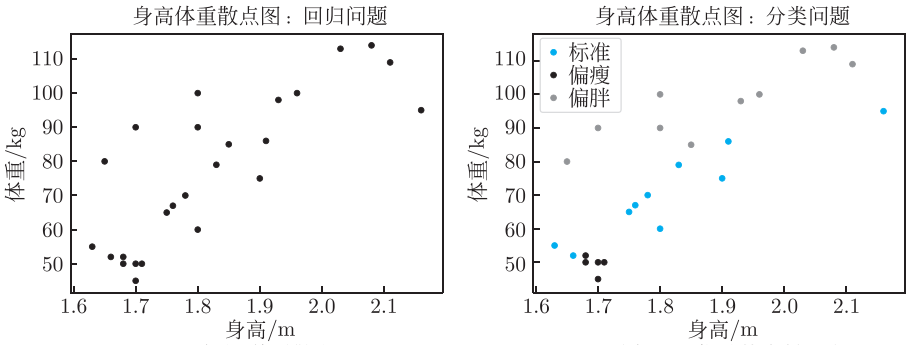
例 1.1.5　(分类与回归)　如图 1.3所示，收集到一些健康成年人的身高与体重信息，横轴表示身高，纵轴表示体重。如果希望描述体重随身高的变化情况，将输入的体重映射到连续的输出，这就对应于一个回归问题；如果希望根据一些衡量身材胖瘦的指标将他们的身材分为标准、偏瘦、偏重 3 个类别，将输入的体重对应到离散的 3 个类别，则对应于分类问题。

为了有效评价有监督学习模型的性能（如分类正确率），还需要准备一个测试数据集，要求测试数据不能在训练集中出现。后文将看到很多具体例子。

无监督学习（unsupervised learning）

与有监督学习对应的是无监督学习，二者的区别在于无监督学习任务中，数据的标签是“缺失”的，即它是一种没有“老师”指导的学习方式。无监督学习的训练数

① http://yann.lecun.com/exdb/mnist/.



(a) 身高–体重散点图　(b) 依据BMI指数的身材分类图

图 1.3　散点图

据一般表示为

$$\mathcal{D} = \{\boldsymbol{x}_i : \boldsymbol{x}_i \in \mathcal{X}\}_{i=1}^{N}. \tag{1.2}$$

无监督学习的目的一般是发现数据中紧致的“隐含”规律，增强对数据的理解。进一步，无监督学习可以分为多种具体的任务，包括聚类、降维、密度估计、表示学习等，下面简要介绍。

(1) **聚类**：聚类的目标是将给定的数据 $\mathcal{D}$ 进行划分，将相似度高的数据划分到同一类，将相似度低的数据划分为不同类。例如，在一次长途旅行中，我们会拍很多漂亮的照片，但浏览时可能感觉杂乱无章、存在重复等，此时，聚类算法可以帮我们将类似地点或类似风景的照片自动做一个归类，以方便后续浏览或查询（详见第 8 章）。

(2) **降维**：在现实场景中，数据通常分布在高维空间中，且呈现“稀疏”的特点，例如，一张分辨率为 300×300 像素的人脸图片，如果直接用像素值表示的话，将是一个 90000 维的向量！又如，对于文本数据，如果采用词袋模型，每个文档是 V 维向量，对于常见的应用，V 可能达到上万甚至十万，但每个文档出现的文字个数往往远小于 V，导致向量中有很多元素都是 0。在高维空间中，直接学习往往面临“维数灾”的问题，且计算和存储代价往往随维度增加而增加。降维的目的是通过降低特征的维度，得到一个紧致的低维特征表示，便于存储或进一步进行数据分析。降维往往也能起到抑制噪声，提高信噪比的作用（详见第 9章）。

(3) **密度估计**：根据给定的观察数据，估计产生数据的未知概率分布。例如，截至 2020 年 2 月 17 日，我国某省共发现 339 例流感确诊病例，年龄介于 1~90 岁。据统计，年龄段分布为 10 岁以下 9 例，11~20 岁 16 例，21~30 岁 39 例，31~40 岁 59 例，41~50 岁 63 例，51~60 岁 64 例，61~70 岁 67 例，71~80 岁 15 例，81 岁以上 7 例。我们可以用直方图更直观地表示不同年龄段患者人群的分布，如图 1.4所示。通过机器学习方法估计数据的概率分布 $p(\boldsymbol{x})$ 是很多任务的基础，比如利用深度概率模型 [4–5] 可以有效刻画高维图像数据的概率分布，并生成高质量的新图片。关于密度估计的更多内容，详见第 4 章和第 16 章。

(4) **表示学习**：与降维紧密相关，表示学习的目标是学习一个变换函数，将原始

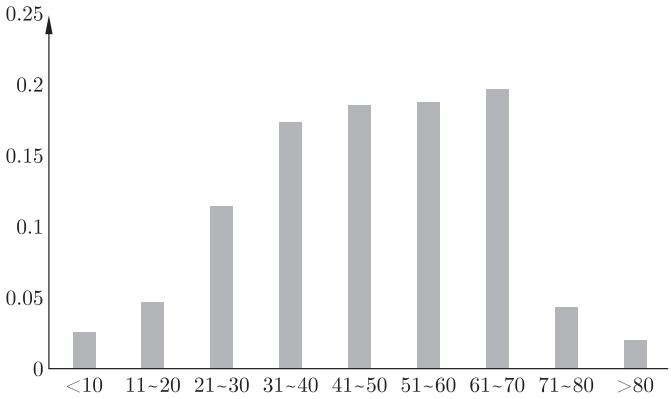


图 1.4　某省流感患者在不同年龄段的分布图

数据映射到一个新的特征空间中，使之具有某些良好的特性，例如，描述单词或文档的语义信息、表示图像数据的低维特征向量等。随着深度学习的进展，具有多层次结构的表示学习方法成为处理图像、文本、语音、图结构等很多种类型数据的首选方案（详见第 6 章）。

值得注意的是，上述任务并不都是无监督的；在有监督的学习任务中，往往也会包含上述这些基本任务，例如，在做图像分类时，适当挖掘数据中的聚类信息往往能提升分类的效果；另外，利用深度神经网络进行图像分类时，很自然地也包括了表示学习。

强化学习（reinforcement learning）

除了上述任务，机器学习中还有很多任务属于决策，其目标是希望习得一种“策略”，能够让决策的收益最大化。例如，中午我们要选择餐厅吃午饭，目标是希望吃到合口的菜肴。一种选择是去曾经去过的最好的餐馆；与此同时，我们还需要兼顾（探索）一些新开的餐馆，因为新餐馆可能有意想不到的、更好的选择。又如，在下棋过程中，我们的目标是希望找到一种策略，能够更高概率地战胜对手。强化学习是解决这类决策任务的有效途径，是区别于有监督和无监督学习的另外一类基本学习任务。

稍微正式一些，强化学习假设环境中有一个智能体，通过观察环境的状态，并采取行动与环境进行交互，获得环境给的收益反馈。智能体的目标是通过多次交互，搜集到经验数据，进行学习，最终目标是找到一个最佳的策略，使得累积的收益最大化。强化学习与环境的交互如图 1.5所示，其中第 t 轮交互时，智能体根据当前状态 S_t，采取行动 A_t，获得奖励 R_t，与此同时，交互使得环境状态从 S_t 转移到 S_{t+1}。由此可见，强化学习与有监督学习或无监督学习有很大不同，后者假设事先给定一个搜集好的数据集，而强化学习的数据是在交互过程中不断搜集的。第 17 章将具体介绍强化学习。

除上述 3 种基本的机器学习任务外，还有半监督学习——部分数据有标注信息而

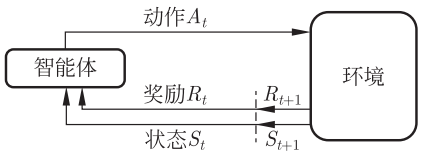


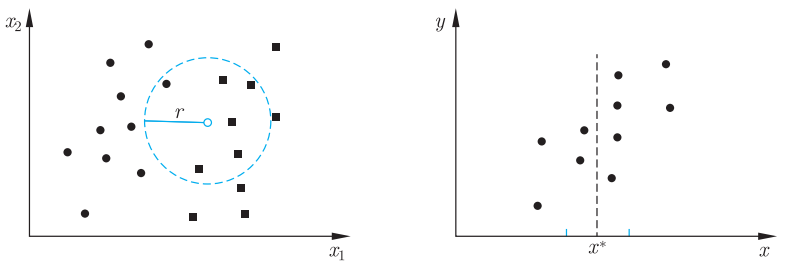
图 1.5　强化学习与环境的交互

大部分数据缺少标注信息，主动学习——从数据中选择一部分进行数据标注，迁移学习——从另外一个领域迁移一些知识到已有数据的领域等。虽然具体任务多种多样，但本书介绍的基本原理和方法具有通用性，可以在不同任务中找到相应的变种。

1.1.3　K-近邻：一种“懒惰”学习方法

每种学习任务下又发展了多种算法。这里举一个用于有监督学习的算法——K-近邻法。它具有简单、易于理解的特点，同时适用于分类和回归任务，且在实际数据集上通常表现良好的性能。

如图 1.6所示，K-近邻法用于分类问题时，采取了很直观的策略：将新的数据点（也称为实例）与训练数据集所有实例进行比较，选出前 K 个最相似的作为分类的依据；其中的相似度则通常由特征空间中的某种距离衡量，距离越小相似度越高。在选取相似数据点时，有多种策略可以选择，例如，算法可能采取将距离从小到大排序后取前 K 个；或者取以新实例为中心一定半径的球内的样本。



(a) 用于分类的K–近邻法　　(b) 用于回归的K–近邻法

图 1.6　K-近邻法

在相似点选取完成后，K-近邻法根据这 K 个训练数据提供的信息对新实例进行分类。常见的分类规则是将新实例分入被选取的实例中多数实例所在的类别，即“多数表决规则（majority voting）”。图 1.6中，圆形范围内的几个训练样例中正方形较多，于是蓝色新数据点被分类为正方形代表的类别。

K-近邻法简单、易于实现。令人惊奇的是，在训练样本足够多时，K-近邻法具有足够强的理论特性。设训练集的规模为 N，使用 K-近邻法进行分类的测试错误率记为 R_{KNN}，理论最小分类错误率记为 R^*（当数据的分布已知时，贝叶斯分类器的错误率是所有可能分类器中最小的，称为贝叶斯错误率，详见第 4 章），则存在如下

不等式[6]：

$$R^* \leqslant \lim_{N\to\infty} R_{\mathrm{KNN}} \leqslant R^*\left(2 - \frac{M}{M-1}R^*\right), \tag{1.3}$$

其中 M 为类别的个数。由此可见，K-近邻法的错误率不超过最小错误率的两倍。

在回归问题中，算法需要根据从训练集获取的信息将新的数据点赋予一个实数类型的预测值。如图 1.6(b) 所示，横坐标代表特征空间中的坐标 x，纵坐标代表实数预测值 y，K-近邻法以一定的标准找到与测试数据 x^* 在特征空间中最相近的若干训练数据，以它们的 y 值加权作为预测值。具体地，式 (1.4) 是 K-近邻法用于线性回归时加权预测公式的一个例子：

$$\hat{y} = \frac{\sum\limits_{i\in\mathcal{N}_k} w_i y_i}{\sum\limits_{i\in\mathcal{N}_k} w_i} = \sum_{i\in\mathcal{N}_k} r_i y_i, \tag{1.4}$$

其中，$\mathcal{N}_k$ 表示最近邻，例如，$\mathcal{N}_k = \{i | d(x_i, x) < r\}$ 表示选取以测试数据为中心半径 r 的球内的数据点，$d(x_i, x)$ 代表特征空间中两点 x_i, x 的距离；$w_i = \dfrac{1}{d(x_i, x)}$ 表示测试数据与训练数据之间的相似度（与距离成反比）；$r_i = \dfrac{w_i}{\sum\limits_{i\in\mathcal{N}_k} w_i}$ 是归一化的权重。

K-近邻法易上手、理论性能好的特点是当今仍广泛应用的原因，但这种方法也存在以下挑战。

(1) K-近邻法对每个测试数据，均需要计算与所有训练数据的距离（相似度），以选择最近邻，计算复杂度为 $O(dN)$，其中 d 为特征维度。对大规模数据集而言，计算资源开销极大。为了提高计算效率，学者们提出了多种加速的近邻搜索算法，例如，基于 k-d 树等数据结构的分支定界搜索算法[7]以及多种近似搜索算法[8–9]等。

(2) K 值的选择与数据集有关，可能会极大地影响模型效果，需要进行调参。K 值较小时模型对近邻的数据十分敏感，易受噪声影响，出现过拟合现象（具体概念在后文介绍）；K 值过大则会使较远的、不相似的数据发挥作用，影响预测，例如，极端情况下设 $K = N$，此时模型会简单地将任一测试数据预测为训练样本中最多的类，使得大量有用信息被忽略。

(3) 数据的表示以及距离度量的选择会对结果产生重要影响。在实际问题中，不同特征维度表示的物理含义或量纲可能不同，如描述一个人的特征可能有身高、体重、年龄、爱好等，对不同类型的特征适当进行归一化再计算距离，往往能显著提升性能；此外，自动学习合适的距离度量（称为距离度量学习，distance metric learning）（更多内容可参阅文献[10]）往往也能显著提升 K-近邻法的性能。

总体上，K-近邻法是一种“懒惰”的学习方法——在学习时，“死记硬背”训练样本；在测试时，通过简单的查询完成预测。与之不同，很多学习方法属于更加“积极”的学习，通过对问题做一些合理的假设（例如，用一个概率分布描述随机掷硬币的观察值；或者用一个线性平面将不同类型的图片进行分类），学习一个更加紧致的