第1章 机器学习基础

1.1 AI for Science 世界的寻宝图

本节重点

1. 理解什么是 AI for Science。
2. 理解促进科技发展的几大范式。
3. 理解 AI 能够应用到科学研究的本质原因。
4. 理解 AI 的基础概念，包括监督学习、无监督学习、分类和回归等。

1.1.1 什么是 AI for Science

当今，新能源材料的研发就像在茫茫大海里捞针。传统方法需要耗费数月甚至数年时间评估一种材料的性能，严重拖慢了固态电池等关键技术的突破。2023 年 11 月底，谷歌的 DeepMind 团队开发了用于材料科学的人工智能强化学习模型 Graph Networks for Materials Exploration(GNoME)，寻找了 38 万余个特性稳定的晶体材料，这大大加快了新材料的研发进度。

事实上，材料科学仅是受益于人工智能(AI)发展的众多科学学科之一。AI 的崛起正在引领科学研究的一场激动人心的转型，其影响已经从实验室扩展到我们每个人的生活中。我们期待一个由 AI 驱动的未来，在这个未来中，AI 工具会将我们从烦琐、乏味和耗时的劳动中解放出来，同时引导我们进行创新性的发明和发现，使原本需要数十年才能实现的科学突破得以提前到来。我们将这一愿景称为“AI for Science”(后简写为 AI4S)。当前，AI4S 已经展示了一系列令人振奋的应用成果。2024 年诺贝尔化学奖得主德米斯·哈萨比斯(Demis Hassabis)和约翰·M. 贾姆珀(John M. Jumper) 的工作就是一个很有说服力的 AI4S 应用实例，他们共同开发的 AI 模型 AlphaFold2 成功实现了对几乎所有已知蛋白质序列的三维结构预测[1]。这一成就解决了生物化学领域长达五十年的核心难题，极大地加速了科学研究的进程。我们坚信，AI4S 已经并将持续深刻改变人类的未来。

相信阅读这本书的读者一定对探索 AI4S 充满兴趣，为此我们“绘制”了两张 AI4S 世界的藏宝图。本节主要介绍 AI4S 的第一张藏宝图，使读者了解 AI4S 由来，了解为什么 AI 能够助力科学研究。我们将在本书的最后一章介绍 AI4S 的第二张藏宝图，使读者了解 AI4S 要到哪里去。

1.1.2 AI4S 第一张藏宝图：AI 求解高维函数

AI4S 的第一张藏宝图是关于 AI4S 的由来，即为什么 AI 能够帮助解决科学问题。想要理解这个问题，就需要了解在 AI 时代之前科学研究是什么样子，又遇到了哪些问题。

1.1.2.1 前 AI 时代科学研究遇到的问题

科学研究是人类文明进步的根本推动力。自文艺复兴以来，科学研究总体上沿着两大范式前进，一种是数据驱动的范式，一种是机理驱动的范式。

数据驱动的范式主要通过分析数据、寻找科学规律来解决实际问题。一个比较经典的采用数据驱动范式的工作是开普勒定律，如图 1.1.1 所示。通过分析和总结丹麦天文学家第谷·布拉赫（Tycho Brahe）提供的精确的天文观测数据，约翰内斯·开普勒（Johannes Kepler）发现了著名的开普勒定律。开普勒定律的发现过程主要依赖于开普勒对数据规律的挖掘，其中开普勒第三定律“行星轨道的半长轴的立方与其公转周期的平方成正比”就是对某种数据关系的总结。由于开普勒的工作非常经典，数据驱动的范式也常被称为开普勒范式。

图 1.1.1 数据驱动与模型驱动

直观上，数据驱动的研究范式很简单，研究者只需要分析数据之间的规律就能找到一些有用的结论。不可否认的是，数据驱动的范式确实在诸如经济学的很多研究领域被广泛使用，但如果想要将其应用到更多科学研究中，科学家们还面临很多问题。一个比较突出的问题是数据稀缺，数据量不足以支撑分析需求。由于数据采集的成本非常高，在绝大多数科学研究场景中，数据是相当稀缺的。材料领域的许多数据主要依赖传统实验获取，而这类实验往往耗时较长且成本巨大，在一些研究领域，全球累计的可靠数据常常不足几十条，远未达到一些先进数据分析方法对数据规模的基本要求。与此同时，在面对海量数据时，现有的数据分析手段又无法进行精细且有效的分析，只能通过类似微积分、线性代数等方式对数据进行相对粗粒度的分析，且缺乏更加精细化的数据分析方式。

机理驱动的范式是一种基于第一性原理的研究方式，它通过揭示现象背后的科学原理来推动人类的进步。应用这种方法的一个典型代表是牛顿运动定律。和开普勒纯粹基于数据分

析的方式不同，牛顿更关心现象背后潜在的科学原理，其发现的牛顿运动定律一度被认为是支配现实世界的科学规律。如图 1.1.2 所示，因此机理驱动的范式往往也被称为牛顿范式。

图 1.1.2　开普勒与牛顿

机理驱动的范式长期以来一直是物理研究的主要范式，物理学家们利用机理驱动范式成功建立了从相对论到量子力学的一系列物理模型。在微尺度领域，随着量子力学理论的建立和完善，人类逐步掌握了解释微观现象的理论工具。正如保罗·狄拉克(Paul Dirac)所说："有了量子力学，除一些极端尺度下的情形，我们已经掌握了大多数工程和自然科学所需要的基本原理。"遗憾的是，研究者后来发现，基于薛定谔方程求解粒子运动的计算成本过高，在绝大多数场景并不适用。

为了解决计算成本过高的问题，物理学家根据不同的时间和空间尺度，发展了从微观、介观到宏观的一系列物理模型，这种方法被称为多尺度建模，如图 1.1.3 所示。多尺度建模在某种程度上已经成为人类现代科学计算的基础逻辑：在处理实际问题时，首先确定问题所涉及的时间和空间尺度，然后选择适当的模型进行计算。然而，在许多情况下，科学家们仍然不得不在计算效率和精度之间做出取舍。比如材料计算中常用的密度泛函理论(density function theory，DFT)虽然精度较高，但通常只能处理几千个原子的计算规模；而基于经验势的分子动力学(molecular dynamics，MD)虽然能够处理高达百万级别的原子计算，但会面临精度太低的问题。在实际应用中，我们迫切需要既精确又高效的算法，但现有的理论往往难以同时满足这些要求。

1.1.2.2　传统计算问题的核心难点：维度灾难

幸运的是，并不是所有问题我们都无能为力，理论计算曾在很多领域为我们提供重要的支持。如图 1.1.4 所示，在新冠疫情初期，国家迅速建设了雷神山医院和火神山医院。为确保医院内的病毒不外泄，建设者们利用"达索"系统模拟了医院内的污染扩散过程，有效指导了医院的设计与建设。在航空工业中，飞机设计也依赖于计算仿真技术，这些技术能够预测飞机在各种飞行条件下的空气动力学特性和相关飞行性能，为飞机的设计与性能优化提供了重要支持。此外，计算仿真技术在机械工程、电子工程等多个领域都得到了广泛应用，并已取得显著成效。

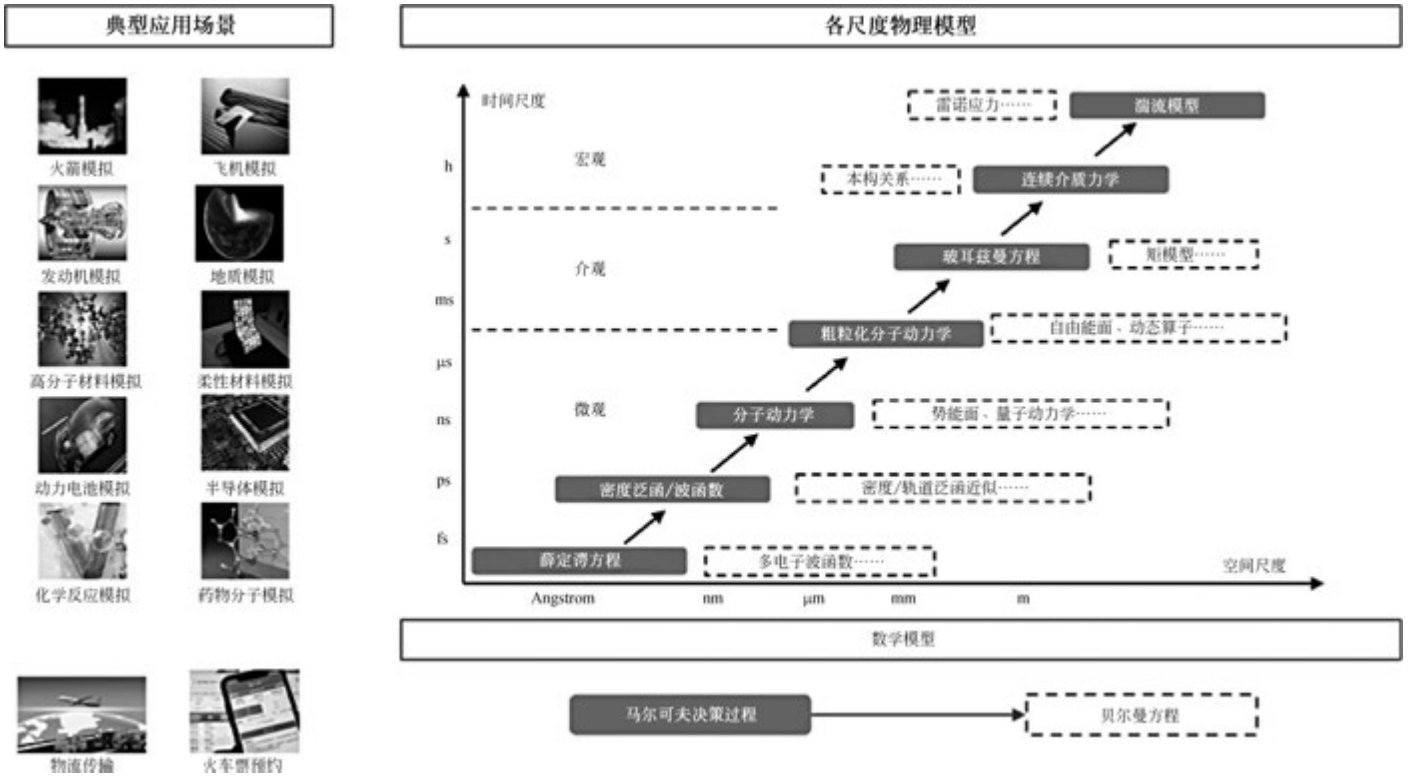

图 1.1.3　应用场景与各尺度物理模型

C919研发使用三维数字样机技术　　雷神山医院污染扩散仿真

图 1.1.4　计算仿真技术成果

计算仿真技术的应用场景通常有如下共同点。其一，处理的往往是一个低维的问题；其二，恰好是计算机擅长解决的问题。比如计算流体动力学(computational fluid dynamics，CFD)的主要研究内容是连续介质的宏观行为。尽管流体的运动是由无数分子的运动组合而成，但在流场计算中，流体通常被看作连续的，一般使用流体力学方程(如纳维-斯托克斯方程)来描述。这种连续介质的假设大大简化了计算，使得我们只需解决速度场、压力场等几个宏观变量的计算，而不必考虑单个分子运动的计算。所以这个问题本质上是一个低维的问题，并且目前已经有了很多成熟的数值解法。

在微尺度领域，这些条件往往不被满足。以分子动力学为例，进行分子模拟就是一个典型的高维复杂问题。该算法需要考虑分子间的每一种相互作用，即便是只包含 100 个原子的体系，其维度也能达到 300 维(每个原子具有 3 个空间维度)。此外，分子动力学需要在飞秒级别的极短时间步长内模拟物理过程的长期演变，这对计算资源和计算效率的要求极高。

对于这样的问题，有一个专业的术语叫作“维度灾难”。维度灾难(curse of dimensionality)是指在处理高维数据或高维空间问题时，随着维度增加，问题的复杂性和计算量呈指数级增长，导致传统方法在高维空间中变得不再适用。从计算机算法的角度，这种问题的计算复杂度是 $O(n^d)$，其中 n 为问题的规模，d 是一个大于 1 的常数。

1.1.2.3　人工智能技术的核心能力：AI 求解高维函数

近年来人工智能技术的快速发展给求解高维问题带来了希望。本书的后续章节将为大家详细讲解人工智能相关技术的原理。当前阶段，希望大家能先从直观层面理解，为什么人工智能有助于解决那些曾长期困扰科学家的问题。

来看两个过去人工智能领域的成功应用。

第一个应用是围棋机器人 AlphaGo。围棋需要棋手在一个 19×19 的棋盘上选取一个最优位置。如果将这个问题视为一个随机落子问题，也就是说棋盘上每个位置有空、黑子、白子三种情况，那么该问题可能的求解空间的维度将会高达 $3^{19\times19}$，该问题是一个高维问题。正是由于这个原因，过去人们一直认为围棋是人类智慧的象征，而计算机很难求解如此复杂的问题。然而，AlphaGo 在 2017 年击败了人类围棋冠军柯洁，成功解决了这一问题。其解决问题的基本思路是，首先通过人类棋谱数据训练策略网络，学会模仿人类高手的下法；之后通过自我对弈，进一步优化策略网络和价值网络，使得模型能够在自我竞争中不断提高。

第二个应用是图像识别。计算机存储图像一般采用矩阵或者张量的形式，一个传统意义上 1024×1024 的彩色图像通常以 3×1024×1024 这样的张量进行存储。对于一个图像识别问题，比如识别某图像是猫还是狗，以数学的观点来看，本质上是找到从图像数据（某个高维张量）到某个类别（猫或狗）的映射函数。因此，图像识别问题本质上也是一个高维问题。近年来一系列算法，从图像分类算法（如 AlexNet）到目标检测算法（如 Yolo），都可以很好地解决这个问题。

这两个应用只是 AI 诸多成功应用中的一部分，但我们可以从中发现，相比于传统数学手段，AI 在求解高维函数方面表现出极为强大的能力。“维度灾难”问题一直是科学研究中的一个重要挑战，因此将 AI 应用于科学研究、利用其在高维函数求解中的优势来克服“维度灾难”成为一个极具价值的研究方向。这正是 AI4S 旨在解决的问题。

1.1.3 机器学习基本概念

通过前面的讨论，整体理解了为什么 AI 能够助力科学研究。接下来，开始接触本书最重要的内容，了解什么是机器学习。

1.1.3.1 什么是机器学习

要理解什么是机器学习，我们可以首先思考一下人类是如何进行学习的。人类一般通过观察现实生活中的现象总结规律，然后内化为自己的经验或知识；当面对新情况时，人类会套用过去总结出的规律来处理问题。类比人类的学习方式，我们希望计算机也能够从大量复杂的现象中抽象出相关规律，并将这些规律应用到新的问题中。这个过程被称为“机器学习”，而通过“机器学习”得到的“规律”则被称为机器学习模型，如图 1.1.5 所示。

图 1.1.5 机器学习概念图

1.1.3.2 人工智能的发展历史

人工智能（AI）的概念可以追溯到 20 世纪中期，当时数学家、逻辑学家艾伦·图灵（Alan Turing）提出了著名的“图灵测试”，旨在判断机器在与人类的对话中能否表现得与人类无法区分。图灵测试是人工智能研究的重要起点。图灵在其 1950 年的论文“Computing Machinery and Intelligence”[2] 中探讨了机器是否能够思考的问题，并提出了通过对话来测试机器智能的概念。

20 世纪 50 年代，随着计算机技术的发展，人工智能作为一个独立的学科开始成型。1956 年达特茅斯会议上，约翰·麦卡锡（John McCarthy）、马文·明斯基（Marvin Minsky）、克劳德·香农（Claude Shannon）、内森尼尔·罗切斯特（Nathaniel Rochester）等提出了“人工智能”这个术语[3]，并共同探讨了让机器模拟人类智能的方法。该会议被认为是人工智能研究的正式起点。

早期的人工智能研究主要集中在符号主义和逻辑推理上，研究人员相信智能行为可以通过操作符号和规则来实现。符号主义的一个代表性应用是专家系统，当时科学家认为通过将领域相关知识用符号和规则来表示，就可以构建一个优秀的专家系统。这个过程的确

产生了一些特定领域的专家系统，比如用于帮助医生诊断和治疗细菌感染的 MYCIN 系统等。然而，随着研究的进一步深入，人们发现，利用符号和规则来表示知识的工作量极其庞大，且很多知识难以用符号和规则来表示，因此符号主义渐渐不再是人工智能研究的主流方向。

人工智能的另外一个重要方向是统计学。统计学派强调通过概率和统计的方法来处理不确定性和推断问题，并相信智能行为可以通过统计学习模型来实现。在深度神经网络之前，统计学派是机器学习最主流的方向，诞生了包括线性回归、逻辑斯蒂回归、贝叶斯网络、决策树、支持向量机等在内的一系列模型。在小规模数据集上，统计模型往往可以达到最优的预测效果，同时统计学派将严格的数学推导引入机器学习中，为相关机器学习模型的建立和发展打下了坚实的理论基础，使得基于统计的机器学习模型具备较强的可解释性。因此，在包括材料学研究在内的很多科学领域，统计机器学习模型依然非常常见。本书的前半部分会选取这里面最具代表性的模型为大家进行详细讲解。

虽然统计学派在人工智能的发展中占据了重要地位，但近年来机器学习的主流方向还是连接主义，或者说是继承了连接主义的衣钵。连接主义主张通过模拟生物神经网络的工作方式来实现智能行为[4]。与符号主义不同，连接主义不依赖于显式的规则和符号操作，而是通过大量简单的处理单元（如神经元）的连接和活动来处理信息。这一流派为现代神经网络和深度学习的发展奠定了理论基础。事实上，连接主义早在 20 世纪 50 年代就已出现，如 1957 年提出的感知机模型[5]。然而随着 70 年代人们开始发现感知机的局限性（如无法解决异或问题等），连接主义逐渐遇冷。尽管后来连接主义不时会出现一些重要工作，比如多层感知机、反向传播算法等，但是由于统计机器学习模型在小规模数据集上的优越表现，连接主义一直无法成为领域主流。2009 年，来自斯坦福大学的李飞飞教授及其团队创建了涵盖数百万标注图像的 ImageNet 数据集，并基于这个数据集发起了 ILSVRC（ImageNet Large Scale Visual Recognition Challenge）竞赛，以鼓励机器学习研究者探索基于更大规模数据的模型。2012 年，来自多伦多大学的 Geoffrey Hinton 和他的学生 Alex Krizhevsky 、Ilya Sutskever 开发的 AlexNet 模型获得了该比赛的冠军[6]。AlexNet 是第一个真正意义上基于大规模数据集训练而成的卷积网络，它显著降低了分类错误率，AlexNet 的诞生标志着深度学习在计算机视觉领域的崛起。

AlexNet 的问世开启了深度学习的大门，此后人工智能再次进入蓬勃发展的时代。起初，研究的重心集中在图形图像领域，出现了包括 GoogLeNet、ResNet 在内的一系列模型；之后人工智能迅速拓展到视频识别、图像生成、自然语言、语音识别等各个领域，涌现出 U-Net、GAN、Diffusion Model、Transformer 等一系列模型。正是从这一阶段开始，人工智能开始广泛应用于我们的生活中，比如前面提到的人脸识别、语音助手等技术。近年来，以 ChatGPT 为代表的大语言模型的出现，则进一步为“通用人工智能”带来了希望。大语言模型的底层技术依然是深度神经网络，但在 Scaling Law 的加持下，我们看到大语言模型相比于传统模型展现出更多的“智能”，正因此，人们对 AI 赋予了更多的期待。希望在不久的将来，一个具备通用人工智能的机器人能够出现并为人类服务。

1.1.3.3 机器学习整体流程与常见分类

了解了机器学习发展历史，下面来简单看一下机器学习的整体流程：正如上文所说，机

器学习首先需要使用一些已知数据(有时候也称为历史数据)来进行学习,这个学习的过程称为“训练”,而用于学习的数据称为“训练数据”。当机器学习训练好一个模型以后,面对未知的情况(即一些新的数据),就可以使用模型来进行预测(也叫推断)。因此,机器学习又可以理解成基于某些已知特征来预测另外一些未知结果的过程,如图 1.1.6 所示。

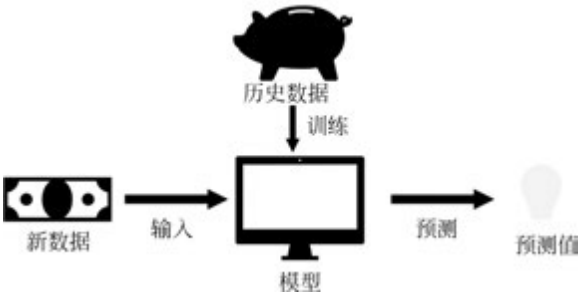

图 1.1.6 机器学习整体流程

机器学习的一个常见分类方式是根据训练数据的特点来划分,通常可以划分为监督学习和非监督学习。假设用于训练的数据已经打好了标签,比如一个基于身高预测体重的模型,训练数据里面既有已知的特征,如身高信息,也有模型要预测的结果,即体重信息,那么在训练过程中,模型就可以知道训练数据的真实预测结果,并通过这个结果逐步改进自己的学习,这就像一个教练时刻拿着标准答案来“监督”模型学习的过程,这样的学习被称为“监督学习”。相反,如果训练数据没有打好标签,比如基于身高来预测性别的模型,训练数据里面只包含身高信息,没有每个身高对应的性别信息,则在训练过程中,模型并不知道真实结果是什么,也就是说没有“教练”拿着标准答案来“监督”,这样的学习被称为“无监督学习”。

需要注意的是,监督学习和无监督学习的区别在于训练过程中数据是否打好标签,而无论什么样的机器学习模型,在实际进行预测(推断)时,所使用的数据都是没有打好标签的(否则就不需要进行预测了)。

机器学习的另一个常见分类方式是按照机器学习的预测目标来划分,具体分为分类问题和回归问题。假设一个机器学习模型最后预测的结果仅有有限的几个类型,它处理的任务就称为分类问题,也可以说该机器学习模型是一个分类模型。例如,猫狗识别是二分类问题,预测材料是否为导体(是和否)也是二分类问题,预测材料稳定性(高温稳定、低温稳定和不稳定)则为三分类问题。如果机器学习模型最后预测的结果是一个连续值,它处理的任务就称为回归问题,也可以说该机器学习模型是一个回归模型。比如预测明天某个股票具体价格、预测材料的电导率、预测某物质的扩散系数等。

最后,我们来区分几个常见的概念,即“人工智能”“机器学习”和“深度学习”。人工智能是一个比较通用的概念,一般来说只要计算机表现出某种程度的“智慧”行为,就可以称其为人工智能。因此,无论是早期的符号主义、统计学派还是连接主义,都属于人工智能的范畴,甚至一些具有简单自动化功能的计算机程序(比如一些传统财务软件等)从广义上来说也可以归类为人工智能。相比之下,机器学习则强调基于数据进行模型训练的过程,其核心在于通过模型对未知数据进行预测。深度学习则是机器学习中神经网络的进一步发

展。一般来说，规模更大、层次更多的神经网络被称为深度神经网络，它的学习过程就被称为深度学习[7]。简单来说，机器学习是人工智能的一种实现方式，深度学习则是机器学习的一种实现方式。

1.1.3.4　机器学习的核心逻辑

了解了机器学习全流程，我们稍微解释一下机器学习的核心逻辑。

如上文所述，机器学习的核心是模型。有时候我们会听到机器学习是一个“黑盒”这样的说法，本质上指的其实是我们使用的机器学习模型是一个“黑盒”。类似“黑盒”这样的描述有时会让初学者产生过度的恐慌和不切实际的神秘感，实际上，“黑盒”仅是在描述当前的深度学习模型往往非常复杂，我们无法明确地解释其内部的作用原理。不过，无论什么样的机器学习模型，其本质都是数学函数，都是从输入数据到输出数据的一个映射。例如，假设想根据晶体的结构来预测晶体的形成能，如果通过机器学习的方式来解决这个问题，会假设存在一个数学函数能够表示晶体结构到形成能的映射，这个函数就是我们希望机器学习模型学习并最终得到的目标。

进一步，为了评估模型学习的效果，机器学习中一般会定义一个“损失函数”，用于刻画机器学习模型和真实函数之间的误差。比如假设机器学习模型用 $f(x)$ 表示，真实函数用 $g(x)$ 表示，那么一个最简单的损失函数就可以用下面这个公式表示：

$$|f(x)-g(x)| \tag{1.1.1}$$

确定了损失函数后，机器学习的目标就变成了找到一个合适的 $f(x)$，使得损失函数尽可能小。理论上我们当然希望损失函数完全为 0，但是这在绝大多数情况下是不现实的。在实际情况中，一般认为当损失函数小于某个阈值时，机器学习模型 $f(x)$ 就可以足够好地表示真实函数 $g(x)$。这也是机器学习模型的一大特点，即模型的目标不是找到精确解，而是一个误差足够小的近似解。

让损失函数尽可能小是机器学习的核心目标，而另一个很大的挑战是真实函数的形式在绝大多数情况下是未知的，即 $g(x)$ 我们是不知道的。比如猫狗分类的算法中，大家并不知道猫狗图像的分类函数是什么形式；晶体形成能预测的任务中，大家也不知道形成能预测的函数是什么形式。幸运的是，人们通常会拥有基于真实函数生成的“数据”，即训练数据。比如在猫狗分类的任务里，人们有大量标注好是猫还是狗的图片，在晶体形成能预测的任务里，人们有很多已知的晶体结构和对应的形成能，这些基于真实函数产生的数据可以帮助我们反推它可能的形式，如图 1.1.7 所示。

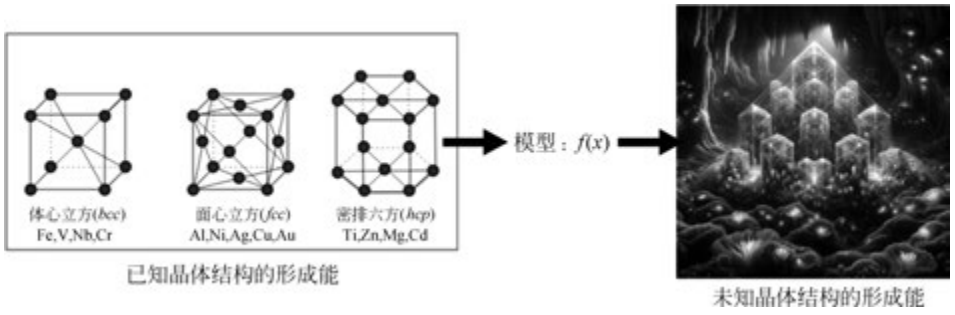

图 1.1.7　晶体形成预测概念图

具体来说，虽然真实 $g(x)$的形式我们并不知道，但是通过训练数据可以知道一些输入的 x 对应的$g(x)$的值，基于此可以进一步细化我们的损失函数，设 X 为训练数据集，N 为 X 集合中元素的数量，则使用均方误差的损失函数可以表示如下：

$$\mathrm{MSE}=\frac{1}{N}\sum_{x\in X}(f(x)-g(x))^2 \tag{1.1.2}$$

除了 $f(x)$的具体形式外，损失函数中的所有值都已知，因此具体值可以求解。接下来训练的过程就是要确定 $f(x)$，使得损失函数值尽可能小，这个步骤一般交给优化算法来完成。

关于机器学习的核心原理就介绍到这里。注意，本节主要是对机器学习相关原理的概述性介绍，旨在让读者对机器学习整体原理有一个大致的认识。关于机器学习原理的具体细节，比如机器学习模型有哪些、常见损失函数有哪些、优化算法原理是什么等，本书后续章节会详细讲述。

1.1.4 怎么使用这本教材

本节的最后我们跟大家聊聊怎么使用这本教材。

与传统更加重视理论的教材不同，《机器学习辅助材料设计》这本教材撰写过程中，除了理论之外，更加重视读者的动手实践。本书的期望之一是，读者在读完本书后，能够具备自己动手编写相关代码的能力，并将机器学习应用到工作和科研实践中。为此，本书每一章都为各位读者精心准备了相关上机案例，这些案例本身甚至比书中的理论内容更有意义。因此，我们不仅希望读者认真学习本书的理论知识，更希望大家仔细阅读这本书中的案例代码，并亲手修改和运行，真正做到“从实践中来到实践中去”。

参考文献

[1] JUMPER J, EVANS R, PRITZEL A, et al. Highly accurate protein structure prediction with AlphaFold [J]. Nature, 2021, 596(7873): 583-589.

[2] TURING A M. Computing machinery and intelligence [J]. Mind, 1950, 59(236): 433-460.

[3] MCCARTHY J, MINSKY M, ROCHESTER N, et al. A proposal for the dartmouth summer research project on artificial intelligence [R]. 1955.

[4] ROSENBLATT F. The perceptron: A probabilistic model for information storage and organization in the brain [J]. Psychological Review, 1958, 65(6): 386-408.

[5] RUMELHART D E, HINTON G E, WILLIAMS R J. Learning representations by back-propagating errors [J]. Nature, 1986, 323(6088): 533-536.

[6] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks [C]//Advances in Neural Information Processing Systems (NeurIPS), 2012: 1097-1105.

[7] GOODFELLOW I, BENGIO Y, COURVILLE A. Deep learning [M]. MIT Press, 2016.

1.2 编程语言基础

本节重点

1. 理解并掌握 Python 语言整体特点。