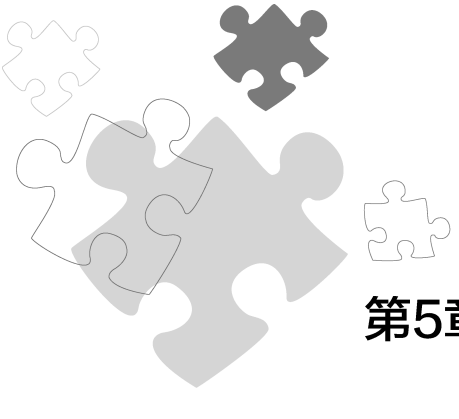


第5章　大数据技术与应用

本章学习目标

- 掌握大数据相关概念、基本特征、思维方式的转变以及数据处理的基本流程
- 掌握多源数据采集方式及数据预处理方法
- 掌握关系数据库的概念和数据模型，了解 NoSQL 数据库和分布式文件系统
- 了解大数据计算
- 掌握数据分析的类型及描述性数据分析的统计指标，了解几种常见的机器学习算法
- 了解数据可视化的作用、典型案例和工具

大数据作为新型的生产资料，正与云计算、物联网、人工智能等技术一起改变着人们的日常生活、企业的生产方式以及人们解决问题的思维方式，大数据已赋能了多个行业和领域。如何高效地产生、收集、处理、存储、计算和分析数据，从大数据中挖掘出价值，完成从“数据”到“知识”与“智慧”的转变，已不只是计算机相关专业学生的“特权”。作为当代大学生，要全面了解大数据，逐步培养自己的数据意识和数据思维。本章对大数据的相关概念、特征、应用等方面进行了概述，梳理了数据流程处理框架，然后依次对数据处理流程中的各环节进行了介绍，包括数据采集与处理、数据存储、大数据计算、数据分析和数据可视化。

5.1　大数据概述

本节将向读者介绍大数据的相关概念和发展背景，分析大数据的基本特征，梳理大数据在多领域的典型应用，分析大数据带来的思维方式的转变，介绍新兴的数据科学学科和数据密集型研究范式，以及梳理数据处理的基本流程框架。

5.1.1　相关概念

1. 数据、信息、知识和智慧

数据(Data)是所有能输入到计算机并被计算机程序处理的符号的总称，是对现实世界的客观记录。数据的外延非常广，可以是数值、文字、图形、图像、语音、动画、视频、社会关系等多种形式的记录。信息(Information)是包含在数据中，能够被人理解的思维推理和结论。

知识(Knowledge)是指从信息中发现的共性规律、模式、理论和方法等。智慧(Wisdom)是运用知识,创造性地进行预测、解释和发现。

例如,在超市购物收银时,顾客购物的时间,物品的单价、数量及总价等都是被客观记录的数据;基于所有顾客的所有购物数据进行分析,得到同时购买啤酒和尿布的数量占所有购物数量的比例是一条信息;在信息的基础上,通过数据挖掘算法发现啤酒和尿布经常被同时购买;根据该信息,对商品进行推荐。再比如,通过测量星球在特定时间的位置,可以获得数据;基于这些数据,可以得到星球的运动轨迹,即信息;通过这些信息总结出的开普勒三定律,就是知识。

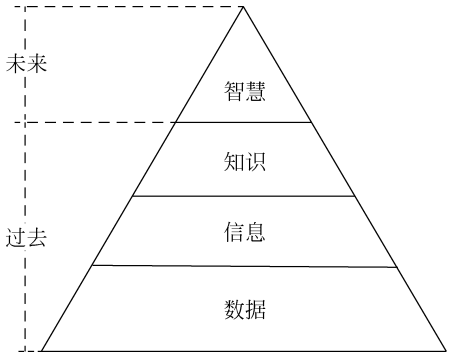


图 5.1　DIKW 金字塔模型

图 5.1 的 DIKW 金字塔模型揭示了数据、信息、知识和智慧间的层次关系,金字塔底层是上层的基础,上层是底层的提升。从“数据”到“智慧”是人们认识程度的提升,也是“从认识部分到理解整体、从描述过去(或现在)到预测未来”的过程。

2. 类别型数据、序数型数据和数值型数据

类别型数据(Categorical)也称为分类型数据,每一个取值都代表了一个类别,如性别的两个取值分别代表两个类别。

序数型数据(Ordinal)和类别型数据的相似之处是每个取值都代表了不同的类别。但是,序数型数据不同的取值既有类别之分,也有大小之分。例如,年收入可以划分为三个等级:高、中、低。

数值型数据(Interval)也称为区间型数据。其取值代表了对象的状态,如年收入的值。

3. 结构化数据、半结构化数据和非结构化数据

根据数据的结构模式的强弱,通常可以把数据划分为结构化数据、半结构化数据和非结构化数据。针对不同的数据,采用的数据管理(如数据存储、数据分析)方法也存在着很大的区别。结构化数据具有很强的结构模式,通常会用不同的属性来描述数据,如可以从姓名、学号、年龄、性别、所在院系等维度来描述学生,如表 5.1 所示。在管理结构化数据时,需要先定义数据的结构,然后按照规定的结构生产、存储或管理数据,即“先有结构,后有数据”。关系数据库专门用来存储结构化数据,具体请详见 5.3.1 节。

表 5.1　结构化数据示例

学号	姓名	性别	年龄	所在院系
S3001	张以	男	18	计算机学院
S3002	赵丽	女	19	管理科学与工程学院
S3003	李静	女	18	信息学院

非结构化数据无法形成统一的描述数据的维度,即难以发现统一的数据结构。在日常产生的大量数据中,非结构化数据占的比重越来越大。例如,文档、图像、音频、视频、存储在文本文件中的系统日志都属于非结构化数据。非结构化数据的存储不能采用关系数据库,通常采用非关系型数据库或者分布式文件系统。

半结构化数据的结构模式的强度处于结构化数据和非结构化数据之间，通常“先有数据，后有结构”。半结构化数据虽然没有预先定义的数据结构，但是有明确的数据标签，用来分割实体和实体的属性，因此通过处理转换后可以发现其结构。一般采用 HTML、JSON、XML 等标记语言表示的都是半结构化数据。图 5.2 为 XML 和 JSON 格式的数据示例。

```
▼<note>
   <to>George</to>
   <from>John</from>
   <heading>Reminder</heading>
   <body>Don't forget the meeting!</body>
 </note>
```

```
{
"employees": [
{ "firstName":"Bill" , "lastName":"Gates" },
{ "firstName":"George" , "lastName":"Bush" },
{ "firstName":"Thomas" , "lastName":"Carter" }
]
}
```

图 5.2　XML(左)和 JSON(右)格式的数据示例

4. 大数据的概念

目前，还没有形成对大数据统一公认的定义，现有定义主要从“现有技术无法处理”和“数据特征”两个维度出发，能被接受的大数据的定义包括如下几种。

维基百科(Wikipedia)定义大数据为规模庞大、结构复杂、难以通过现有商业工具和技术在可容忍的时间内获取、管理和处理的数据集。

麦肯锡全球研究机构(McKinsey Global Institute)认为大数据是大小超过经典数据库软件工具收集、存储、管理和分析能力的数据集。

徐宗本院士认为大数据是不能集中存储并且难以在可接受时间内分析处理，其个体或部分数据呈现低价值性，而数据整体呈现高价值的大量复杂数据集。

美国国家标准技术研究院(NIST)认为大数据是由具有规模巨大、种类繁多、增长速度快和变化多样，且需要一个可扩展体系结构来有效存储、处理和分析的广泛的数据集组成的。

5.1.2　大数据发展背景

IT 领域经历了三次信息化浪潮，如表 5.2 所示。1980 年前后，个人计算机开始普及，计算机走入千家万户，人类迎来了第一次信息化浪潮；1995 年前后，人类开始接触互联网，人类迎来了第二次信息化浪潮；2010 年前后，大数据、物联网和云计算快速发展，大数据时代已经到来。

表 5.2　三次信息化浪潮

信息化浪潮	发生时间	标　志	解决的问题	代表企业
第一次浪潮	1980 年前后	个人计算机	信息处理	Intel、AMD、IBM、苹果、微软、联想等
第二次浪潮	1995 年前后	互联网	信息传输	谷歌、雅虎、阿里巴巴、百度、腾讯等
第三次浪潮	2010 年前后	大数据、物联网和云计算	信息爆炸	—

大数据时代的到来主要有两个方面的原因：一方面信息技术的快速发展，为大数据时代的到来提供了技术支撑，主要包括信息存储设备容量不断增加，信息处理能力大幅提升以及网络带宽不断增加，这些使得信息传输更加顺畅；另一方面，数据产生的方式也发生了变

革，进一步促进了大数据时代的到来。生产数据的方式经历了运营式系统阶段、用户原创内容阶段与感知式系统阶段。在运营式系统阶段，数据的生产是被动的，往往会伴随着实际的企业业务发生，只有这时才会有新的数据产生；在用户原创内容阶段，互联网的快速发展迎来了以“用户产生内容”(User Generated Content，UGC)为特征的 Web 2.0 时代。相对于 Web 1.0 时代以门户网站显示信息为主，Web 2.0 时代以博客、微博、微信等产品为代表，用户可以随时随地产生数据，且以非结构化数据为主；在感知式系统阶段，物联网中大量的传感器，如温度传感器、湿度传感器，每时每刻都在产生数据。

5.1.3 大数据的基本特征

通常认为，大数据的基本特征可总结为“4V”，即规模庞大(Volume)、多样性(Variety)、时效性(Velocity)和价值大但价值密度低(Value)。

1. 规模庞大

相对于现有的数据存储和计算能力，普遍认为 PB 级的数据就可以称为“大数据”。目前已形成了“大数据摩尔定律”，即全球的数据量正以每 18 个月至 24 个月翻一番的速度快速增长。据互联网数据中心(Internet Data Center，IDC) 统计，预计到 2025 年，全球数据总量将达到 163ZB。数据规模的不断增长，必然会对数据的获取、传输、存储、处理和分析带来挑战。

2. 多样性

在大数据中，多种类型的数据往往共存着，包括结构化数据、非结构化数据和半结构化数据。据统计，在未来，非结构化数据的占比将达到 90%以上。例如，在智慧交通这一应用领域，涉及的数据包括结构化的车辆注册数据、驾驶人基本数据、城市道路数据等，也包括非结构化的交通路口摄像头数据等。

3. 时效性

时效性是指数据刻画的事物状态是动态的，是在不断地、持续地发生变化的。因此，大数据应当具有持续的数据获取和更新能力，这对大数据的处理时间要求越来越高。即在某些场景下，如在处理交通路况信息时，要满足数据的时效性要求。

4. 价值大但价值密度低

价值大是指在大数据的基础上，应用数据挖掘、机器学习等技术，可以获取蕴含在数据中，非显而易见的高价值信息或知识。价值密度低是指对于一个特定的应用场景，大数据中真正“有用的”数据是很少的，大量的数据都与目标任务无关。因此，给定具体的任务，如何从大量数据中快速地定位“有用的”数据是大数据计算的核心问题之一。

在大数据“4V”特征的基础上，又增加了一个新的特征即真实性(Veracity)，形成了大数据的“5V”特征。真实性特征强调了数据质量对大数据发挥价值的重要作用，一方面要对数据中的各类噪声数据、缺失数据进行处理；另一方面也要保证数据是客观世界的反映，避免虚假、错误数据的影响。

5.1.4 大数据的典型应用

1. 金融行业

金融行业是大数据应用的前沿领域，大数据在该行业的客户关系管理、股价预测、信贷

风险管控、高频交易等方面都发挥着重要的作用。以股价预测为例，传统的股价预测会考虑风险、收益和企业的状况，然而市场情况对金融市场也有着重要的影响，是预测股价的一个新视角。2011 年 5 月，英国对冲基金 Derwent Capital Markets 建立了 4000 万美元的对冲基金，该基金通过分析 Twitter 的数据内容来捕获市场情况，进而指导投资。利用 Twitter 的对冲基金在首月的交易中以 1.85%的收益率盈利了，而其他对冲基金的收益率平均值只有 0.76%。麻省理工学院的研究者，把 Twitter 上的内容分为了正面或负面情况。经过研究发现，无论是正面情况（如“希望”），还是负面情况（如“害怕”“担心”），它们占总 Twitter 内容数的比例都与道琼斯指数、标准普尔 500 指数、纳斯达克指数的下跌相关。美国佩斯大学的一位博士采用了另外一种方法研究社交媒体数据对股价的影响，他追踪了可口可乐、星巴克和耐克三家公司在社交媒体上的受欢迎程度，并比较它们的股价。通过研究发现，Twitter 上的用户数、Facebook 上的粉丝数和 YouTube 上的观看人数都和股价有密切的关系；并且品牌的受欢迎程度还能帮助预测 10 天、30 天后股价的上涨情况。

2. 会计行业

传统的会计学强调三张报表：资产负债表、现金流量表和利润表，分别反映企业的运营能力、偿债能力和盈利能力。但对于某些类型的企业，如长周期、高负债、高不确定性的 IT 企业、新行业企业、创业企业，它们的无形资产（如客户忠诚度、口碑和品牌）对于衡量企业真正的价值可能更为重要，传统的三张报表就显得捉襟见肘。因此，会计业界和学界提出“第四张报表”来反映相关的数据资产。由于财务数据是对企业过去经营结果的静态记录，因此无法及时反映企业的业务变化；而企业的业务数据，如大量的用户特征数据、用户交易记录、用户偏好、用户对产品的使用行为都是动态的，可能会更及时地反映企业的当前价值。因此，德勤（Deloitte）建立了“业务数据—财务表现—价值评估”的价值评估模型，提出的“第四张报表”强调以非财务数据为核心，以企业绩效为基础，关注数据资产价值，以期为企业提供更全面的价值评估和更深入的管理洞见。

3. 商业

商业领域是大数据发挥价值最多的行业之一。当你去互联网上购物时，一些网站能够做到“千人千面”，向不同的用户推荐不同的商品，这背后就是大数据在发挥作用。网站会根据用户的浏览行为和购买行为，推断出用户的兴趣偏好，并匹配到类似用户的行为，从而为用户推荐他们可能感兴趣的商品。据估计，亚马逊销售额有 1/3 是靠给用户推荐而产生的。《纽约时报》在 2012 年报道了美国第二大连锁百货店塔吉特应用大数据的案例。塔吉特工作人员经过数据分析发现，女性在怀孕不同阶段购买的物品呈现很大的相似性。因此，根据顾客购买的物品可以预测女性怀孕的概率。例如，一位女性购买过大瓶椰子油润肤露、一个大挎包、维生素和鲜亮的地毯，那么就可以估计出她怀孕的可能性是 87%。如果预测出女性怀孕了，塔吉特就会在孕妇怀孕的不同时期向她们推送精挑细选的 25 类商品的优惠券。

4. 生物医药

基于大数据分析，可以实现流行病预测、智慧医疗和健康管理。

1）谷歌流感趋势预测（Google Flu Trend，GFT）

传统的公共卫生管理中，预测疾病流行趋势主要依赖于患者去医院就诊后，医生上报给疾病控制与预防中心。疾控中心基于各级医疗机构上报的数据，发布流行病趋势预测报告。但这种方式一般会有 1～2 周的滞后期：一方面感染人群往往会在发病比较严重后才会到

医院就诊；另一方面，疾控中心需要对医生上报的数据进行汇总与分析。而两周内疫情可能早已扩散。2009年谷歌的科研人员在《自然》杂志上发表论文，从2003—2008年季节性流感传播期间网民在谷歌搜索引擎中输入的4.5亿关键词中挑选出了45个重要的检索词条和55个次重要词条，与同时间段的疾控中心发布的感染人数构建回归模型。在2009年冬季流行感冒预测任务中，与官方数据相比，预测准确率高达97%，并且相对于官方其预测及时性更强。

2）智能疾病诊断

对于利用人工智能（Artificial Intelligence，AI）进行疾病诊断，也需要以大数据为基础。例如，吴恩达所在的斯坦福实验室团队基于目前最大的X光数据库ChestX-ray14数据集（包含来自3万多位患者的超过11万张正面胸片），训练了一个X光诊断算法，可以诊断14种疾病，如肺炎、胸腔积液、肺肿块等。最终，在其中10种疾病的诊断上，AI都与人类放射科医生的表现相当，并在一种疾病的诊断上超过了人类，并且AI的诊断速度是人类的160倍。该团队的另外一个工作是基于大量的电子病历数据，预测病人未来3～12个月的死亡率，确定其是否需要临终关怀。

在其他领域，大数据也发挥着重要的作用。在教育行业，可以基于大数据分析对学生进行“隐形补助”，或帮助学生进行个性化学习；在电信行业，可以帮助预测客户流失概率，进行客户细分；在体育行业，《点球成金》的电影展示了数据在体育行业的重要作用；球员运动装备上的传感器、训练场地的摄像头收集到的大量数据也能够帮助球员提高训练效果。在NBA勇士队，主教练科尔根据团队对历年NBA比赛的统计数据，发现最有效的进攻是传球和投篮，而不是突破和扣篮，因此制定相应的训练战略；在制造业，基于多源数据，如物联网数据、内部业务系统数据（如ERP、CRM、MES、PLM）和外部数据，可贯穿企业生产制造、售后服务、研发设计和企业管理等各个环节，应用于现有业务优化，促进企业升级转型；在城市管理方面，能够利用大数据实现智慧交通、智慧政务、城市规划等；互联网行业也是大数据技术应用最为广泛的领域之一，借助于大数据技术，可以分析客户的各种行为，在此基础上进行商品推荐、有针对性地投放广告、预测客户点击率等。

5.1.5 大数据带来的思维模式转变

V. Mayer Schönberger 和 K. Cukier 在论著 *Big data：A revolution that will transform how we live，work，and think* 中提到大数据带来的思维变革主要有以下几个方面。

1. 从随机抽样到尽量收集完备的数据

在过去，数据获取难度大，开展数据分析一般依靠统计学，采用随机采样获得小数据。然而要满足采样数据具有绝对的代表性这一要求非常困难，因此分析结果容易产生偏差。在大数据时代，要求数据或某个领域的局部完备性。鉴于目前各类传感器、网络爬虫等数据收集手段的普及和发展，收集全面和完整的数据成为可能。

2. 从追求数据的精确性到可以牺牲一部分精确性而追求大数据

对于小数据，由于在进行随机抽样时，少量的错误也可能导致比较严重的偏差，因此一般对其精确性要求比较高。对于大数据，保障其精确性难度比较大。一方面，来源于不同数据源容易造成数据不一致；另一方面由于网络等原因，通过传感器、网络爬虫收集的数据经常出现缺失，使得数据不完整。当数据量非常大并且来源广泛时，会缓解数据不精确带来的

影响。

3. 从因果关系到相关关系

基于因果逻辑推断和利用相关关系是分析数据和预测未来的两种常用方法。在过去，人们一直重视因果关系，认为如果没有分析出原因作为基础，得出的结论不能令人信服。一般来讲，新药的研发大多是基于因果逻辑，即首先要找到致病的原因，才能有针对性地找到解决方案，进而合成新药，对于新药的有效性知其然也知其所以然。举例来说，青霉素的发现过程就是符合因果关系的。1928 年，英国医生亚历山大·弗莱明(Alexander Fleming)偶然发现霉菌可以杀死细菌，但并不清楚霉菌杀菌的原理。直到 1939 年，厄恩斯特·钱恩(Ernst Chain)等人发现青霉素可以杀死细菌是由于一种叫青霉烷的有效成分。青霉烷可以破坏细菌的细胞壁，而人和动物的细胞没有细胞壁，因此青霉素可以杀死细菌却不会伤害人和动物。基于此，美国麻省理工学院的科学家约翰·希恩(John Sheehan)成功地合成了青霉素。但是基于因果关系研制新药是非常漫长的过程，并且成本非常高。如今，利用大数据寻找特效药的方法发生了变化。如果将已经存在的药物和每一种疾病进行配对，可能会发现一些意外的效果。例如，斯坦福大学医学院经过研究发现，本来用于治疗心脏病的某种药物对某种胃病的治疗非常有效。这是一种基于相关关系的方式，通过这种方式，时间和经济成本都会大大降低。谷歌流感预测 GFT 也是利用了搜索关键词和流感患病人数之间的相关关系，而非因果关系。当然，应用相关关系的前提是要有足够多的数据。值得注意的是，在大数据时代，因果关系并非不重要，因为其具有很强的可解释性。

5.1.6　数据科学

数据科学是对大数据世界的本质规律进行探索与认识，是基于计算科学、统计学、信息系统等学科的理论，甚至发展出新理论，研究数据整个生命周期的本质规律，是一门新兴的学科。数据科学的发展历史可以追溯到 1974 年，图灵奖得主、丹麦计算机科学家彼得·诺尔(Peter Naur)提出了“数据学”的概念，研究对象是数值化的数据。他认为数据学是计算机科学的延伸。2001 年，贝尔实验室的威廉·克利夫兰(William S. Cleveland)从统计学的角度出发，提出数据科学应为一个从统计学延伸出的独立研究领域。2007 年，图灵奖获得者 Jim Gray 提出了科学研究的第四范式——数据密集型科学，成为继“实验科学范式”“理论科学范式”“计算科学范式”之后的第四范式。实验科学以观察和总结自然规律为特征；理论科学以模型和归纳为特征；计算科学以模拟仿真为特征；而数据密集型科学的特征是以数据为中心，以数据驱动为手段，以跨领域应用为导向，实现从“数据”到“知识”和“智慧”的转换。数据科学已成为与经验科学、理论科学、计算科学并列的科学研究领域。

数据科学的研究范畴主要包括两个方面：①采用数据驱动的方法研究不同领域的科学，即数据密集型科学发现；②用科学的方法研究数据，主要讨论对大数据更有效的管理，包括数据采集、数据存储、大数据计算和分析，涉及统计学、数据库和机器学习等领域。

5.1.7　数据处理的基本流程

数据处理的基本流程主要包括数据采集与治理、数据存储、大数据计算、数据分析与数据可视化。

1. 数据采集与治理

数据采集是支撑大数据上层应用的基础。大数据的来源多样，既可以来源于数据库，也可以来源于各种类型的传感器、智能终端、互联网以及系统日志文件等。这些数据可以是自动产生的，也可以是由人类生产出来的。通过数据采集获取的数据通常不能直接用于后续的数据处理与数据分析，比如数据中含有大量缺失值、噪声数据或不一致数据。因此，需要对数据进行预处理，提升数据质量，为大数据的上层应用奠定基础。具体内容请详见5.2节。

2. 数据存储

对于不同类型的数据，需要选择不同的数据存储方式进行保存。常用的数据存储方式包括关系数据库、分布式文件系统、NoSQL数据库和数据仓库等。具体内容请详见5.3节。

3. 大数据计算

大数据计算是充分挖掘大数据价值的重要手段。大数据的特征尤其是其规模性和对时效性的要求给数据的计算带来了直接的挑战。为了应对这些挑战，分布式计算已经逐渐成为主流。涉及的技术主要包括MapReduce、Storm和Spark等。具体内容请详见5.4节。

4. 数据分析

数据分析的目标是从杂乱无章的数据中发掘有用的知识，以指导人们进行科学的决策。数据分析可以分为四类：描述性分析、诊断性分析、预测性分析和规范性分析。描述性分析和诊断性分析关注的是过去已经发生的，分别关注的是“已发生了什么”和“为什么发生”；预测性分析主要预测未来将会发生什么，如预测店铺未来的销售额，预测未来患流感的人数；规范性分析主要基于运筹学、模拟和仿真技术，解决优化问题。数据分析主要通过统计、机器学习等方法实现。具体内容请详见5.5节。

5. 数据可视化

为了帮助用户更直观、有效地理解和分析数据，可以进行数据可视化，将数据转换成图形图像并提供交互。具体内容请详见5.6节。

5.2 数据采集与治理

大数据可以由多种方式产生，例如，UGC(用户原创内容)数据，通过用户输入的企业运营数据或者通过感知设备生成的数据。这些数据被生产出来之后，需要把它们收集起来才能在其基础上挖掘潜在的价值，正所谓“巧妇难为无米之炊”。在数据被收集之后，一般仍不能直接使用，需要对数据进行处理，主要包括数据集成、数据清洗和数据变换。本节重点介绍多源数据采集和数据的预处理。

5.2.1 多源数据采集

数据收集旨在从真实世界中获得原始数据。企业内部的业务数据一般会随着业务的开展自动积累下来，因此不做详细介绍。本小节将主要介绍四种数据采集的方式，分别为系统日志记录与用户行为数据采集、感知设备数据采集、网络数据采集和与数据机构进行合作。

1. 系统日志记录与用户行为数据采集

系统日志在系统运行过程中自动产生，一般以文件格式进行记录，主要包括系统访问日

志、用户点击日志等。系统日志可有效地帮助用户诊断错误，辅助系统运营，优化系统运行的效率。例如，根据系统访问日志可有效地描述系统的流量、活跃用户数等情况。系统日志记录采集可通过系统日志采集工具（如 Flume）实现。

用户行为数据描述了用户进入系统后进行的操作，如用户在某个页面停留的时间，点击的按钮，将什么商品加入过购物车，将哪些商品从购物车中移除等。通过用户的这些行为数据可以推理用户的偏好，进行用户画像，为用户提供更精准的服务，如推荐用户可能会购买的商品。对于互联网应用，可以通过“埋点”（事件追踪）的方式获得用户行为数据。“埋点”是针对特定用户行为或事件进行捕获、处理和发送的相关技术及其实施过程。

2. 感知设备数据采集

感知设备数据采集是指通过智能终端（如传感器、射频识别技术和摄像头）采集信号、图片或录像，从而获取数据。传感器可以将物理环境变量转换为可读的数字信号，主要包括温度传感器、湿度传感器、压力传感器等，可以用于智能制造（如监控设备运行状态），进行环境监控、水质监测等。传感器是物联网的重要组成部分，可以通过有线传感器网和无线传感器网将采集到的信号上传到互联网，实现万物互联。RFID（射频识别）主要包括三部分：标签（Tag）、阅读器（Reader）和天线（Antenna）。标签由耦合元件和芯片组成，每个标签有唯一的编码；阅读器可以读取标签信息；天线可以在标签和阅读器之间传递射频信号。RFID 技术在仓储/物流、身份识别、零售等领域已被广泛采用。

3. 网络数据采集

对于互联网上的公开数据，理论上都可以通过网络爬虫或调用网站开放的 API（应用程序接口）来进行采集。

网络爬虫是一种机器人程序，可以自动采集多个网页。互联网上的任何网页，都可以经过若干个超链接到达，网络爬虫会从一个或若干个种子 URL（统一资源定位系统）开始，通过一定的搜索策略（如广度优先搜索和深度优先搜索），依次去访问其他网页。目前已经有很多网络爬虫产品，如八爪鱼、神箭手、火车头。这些产品不需要任何编程基础即可使用，入门门槛比较低。如果读者本身有一些编程基础，也可以自己编写程序实现特定的网络爬虫实现网页数据的采集，如使用 Scrapy 框架，随后再将网页中自己感兴趣的数据提取出来。

有些大型互联网公司（如 Twitter、百度地图）会开放应用程序接口（API），用户可以通过相关网站规定的格式进行数据请求，网站服务器会返回相应的数据。

4. 与数据机构进行合作

如果需要使用外部数据，除了收集网络上的数据外，还可以考虑和其他机构进行合作。需要注意的是，机构间进行数据共享之前必须对数据进行脱敏处理。

5.2.2　数据的预处理

数据预处理阶段的工作主要包括数据集成、数据清洗和数据变换。

1. 数据集成

采集阶段的数据可能有不同的来源，其模式和语义可能存在不一致的情况，如某个应用用男/女表示性别，有的应用则用 0/1 表示性别。同时，在大数据分析阶段，需要将多维数据/视图整合起来查看才更能充分发挥大数据的价值。例如，如果能同时了解用户的个人信息与其社交网络信息，对用户的了解就会更加全面。因此，在数据集成阶段，要进行模式匹

配和语义翻译。前者将解决不同来源数据的异构性，如使用不同的模式表达相同的信息或者同一数据代表不同的含义；后者主要实现实体匹配，将不同的表述映射至同一个事物。在数据集成时，也要对冗余数据和冲突数据进行处理。

2. 数据清洗

在数据清洗阶段，主要工作包括补全缺失值、去除冗余数据、识别和去除异常值、发现和解决数据不一致等。

3. 数据变换

由于数据的量纲和范围可能会有差别。为了更好地服务于后期的数据分析，数据变换主要工作包括简单函数变换、数据的标准化和归一化、数据平滑。

例如，时间序列分析可以通过简单的对数变换或差分运算将非平稳序列转换为平稳序列。对于某些数据挖掘算法，会受到不同数据范围的影响。例如，进行客户细分时，客户的收入范围为1000～50 000元，而客户的年龄为1～100，因此在没有进行数据变换前，客户的相似度计算会倾向于收入的影响。数据的标准化和归一化可以解决上述问题，即规范地将数据缩放到同一个特定范围内。为了缓解数据中噪声的影响，可以通过分箱、聚类等方法对数据进行平滑。例如，将客户的年龄段划分成0～12、12～18、18～30、30～60、60以上几个阶段，即使有些客户的年龄不是很准确，也可能会落到同一个“箱”中，不会对后续数据分析造成影响。

5.3 数据存储

数据被采集之后，需要将数据保存下来，并提供有效的数据查询和检索机制。对于结构化数据，传统的数据存储方式为关系数据库。自20世纪80年代以来，关系数据库在学术界和业界都占据着主导地位。但在大数据时代，数据具有体量大、数据类型多样、对性能和效率要求不断提高的特点，传统的数据存储方式已不能满足需求，因此出现了分布式文件系统、NoSQL(Not only SQL)数据库等新型的数据存储方式。简单来说，数据库是按照一定的格式存放数据的仓库。严格来讲，数据库是长期存储在计算机内有组织的可共享的大量数据的集合。数据库中的数据按照一定的数据模型进行组织、描述和储存，具有较小的冗余度、较高的数据独立性和易扩展性，并可为各种用户共享。

本节重点介绍关系数据库的发展、组成，重点讲述数据库的概念模型和关系数据模型，介绍用于用户和数据库管理系统进行交互的结构化查询语言，分析数据库事务应具有的特性，并总结常用的关系数据库产品；简单介绍四种NoSQL数据库，即键值对数据库、列族数据库、文档数据库和图数据库；最后介绍分布式文件系统。

5.3.1 关系数据库

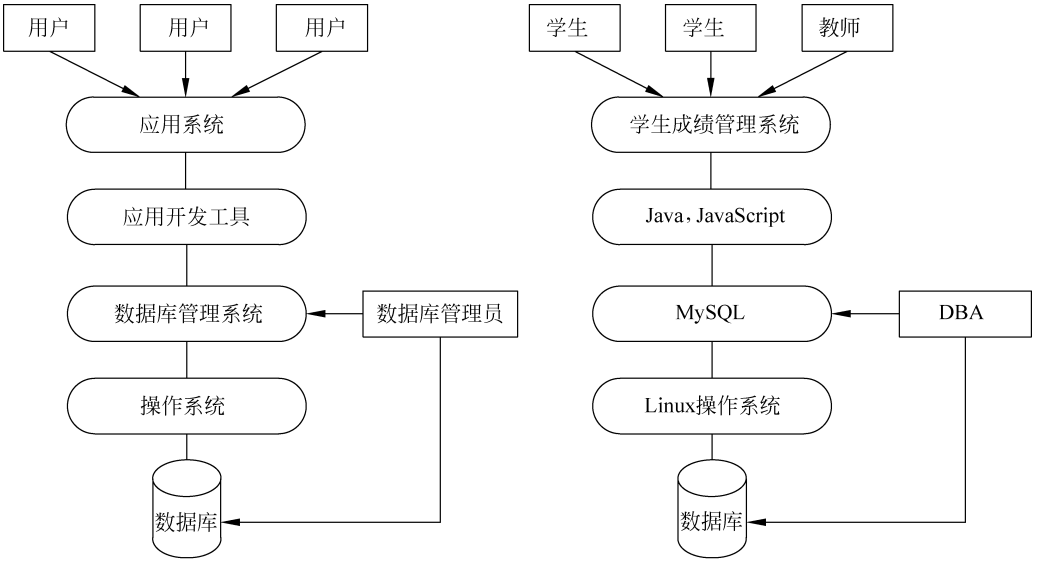
1. 概述

传统的数据存储方式经历了人工管理、文件系统和数据库系统三个阶段。人工管理和文件系统阶段数据的共享性差、冗余度高，且数据独立性差。在文件系统中，一个(或一组)文件对应一个应用程序，即使不同应用程序间有重叠的数据，也不能共享这些数据，而是必须各自创建对应的文件，因此数据冗余性大。这样，一方面会浪费存储空间，另一方面相同

数据重复存储，容易造成数据的不一致，也给数据的修改和维护带来了挑战。为了解决多用户、多功能的数据共享以及数据独立性问题，数据库管理系统应运而生。

数据库管理系统(DataBase Management System，DBMS)是介于用户与操作系统之间的数据管理软件，以具有国际标准的 SQL(Structured Query Language，结构化查询语言)作为关系数据库的基本操作接口。该软件的主要功能包括数据定义功能，即定义数据库中的数据对象的结构；数据组织、存储和管理，即通过多种存取技术(如索引、Hash)提高数据的存取效率；数据操纵功能，即实现对数据库的基本操作，如查询、插入、删除和修改；数据库的事务和运行管理，例如保证数据的安全性、完整性，多用户对数据的并发使用，发生故障后的系统恢复以及数据库的建立和维护功能。

与数据库管理系统有关的另一个概念是数据库系统(DataBase System，DBS)。数据库系统是由数据库、数据库管理系统、应用系统、应用开发工具、数据库管理员和用户组成的存储、管理、处理和维护数据的系统，主要组成如图 5.3 所示。



(a) 数据库系统的组成　　(b) 数据库系统示例

图 5.3　数据库系统的组成及示例

2. 数据模型

数据模型是对现实世界数据特征的模拟，用计算机能够处理的数据描述现实世界中的具体事物。在开发数据库应用系统过程中需要使用不同的数据模型，包括概念模型、逻辑模型和物理模型。概念模型不依赖具体的计算机系统，负责将现实世界抽象为信息世界；逻辑模型和物理模型负责用计算机可以处理的方式将信息世界转换为机器世界。其中，逻辑模型描述了数据的组织方式，主要包括层次模型、网状模型、关系模型、面向对象数据模型等。因为关系数据库系统采用的是关系模型，本小节只对关系模型进行介绍。物理模型描述数据在计算机系统中的存取方式，最终用户是不必考虑数据的物理模型的。

1) 概念模型

现实世界中客观存在的并相互区别的事物在信息世界中用实体表示，实体具有的某一

特性称为属性，不同实体类型间存在着联系，能够唯一标识实体的属性集称为码，同一个类型的实体的集合叫作实体集。例如，一个学生是一个实体，学生的姓名是学生实体的属性，学生实体和课程实体之间存在着学生选修课程的联系，学生实体的码是学号，而不是姓名，全体学生是一个实体集。

实体之间的联系可以划分为一对一联系、一对多联系和多对多联系三种类型。

(1) 如果实体集 A 中每一个实体和实体集 B 中的至多一个(一个或 0 个)实体有联系，反之亦然，则 A 和 B 间的联系类型是一对一联系，记作 1∶1。

(2) 如果对于实体集 A 中每一个实体，实体集 B 中有 $n(n\geqslant 0)$个实体与之联系；反之，实体集 B 中每一个实体和实体集 A 中的至多一个(一个或 0 个)实体有联系，则 A 和 B 间的联系类型是一对多联系，记作 $1:n$。

(3) 如果对于实体集 A 中每一个实体，实体集 B 中有 $n(n\geqslant 0)$个实体与之联系，反之亦然，则 A 和 B 间的联系类型是多对多联系，记作 $m:n$。

概念模型可以用 P. P. S. Chen 在 1976 年提出的实体-联系方法表示，即 E-R(Entity-Relationship)图。在 E-R 图中，分别用矩形框、菱形框和椭圆形表示实体、实体间的联系以及实体的属性。图 5.4 描述了学生选修课程的 E-R 图。

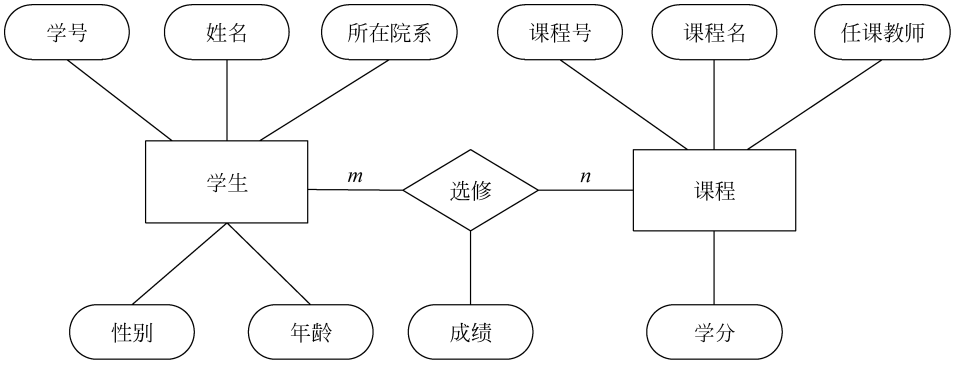


图 5.4　学生选修课程的 E-R 图示例

2) 关系数据模型

数据概念模型梳理完后，要把概念模型转换为数据库管理系统产品支持的逻辑模型。关系数据库系统采用的逻辑模型是关系数据模型。关系数据模型的基本结构为关系，一个关系对应一张二维表，表的名称为关系名。表 5.3 至表 5.5 都属于二维表。二维表中的一行被称为一个元组(Tuple 或记录)；二维表中的一列被称为一个属性(Attribute 或字段)；属性的取值范围被称为域(Domain)；元组中一个属性值被称为分量。一个关系模式可以表示为：关系名(属性 1，属性 2，…，属性 n)。关系中的每一个分量必须是不可分的，如表 5.6 中的基本属性又被细分为性别和年龄是不满足关系模型的基本规范的。

表 5.3　学生表

学号	姓名	性别	年龄	所在院系
S3001	张以	男	18	计算机学院
S3002	赵丽	女	19	管理科学与工程学院
S3003	李静	女	18	信息学院

表 5.4　课程表

课　程　号	课　程　名	任 课 教 师	学　　分
C001	计算机基础	T0001	3
C002	会计学	T0002	2
C003	大学英语	T0003	2

表 5.5　选课表

学　　号	课　程　号	成　　绩
S3001	C001	88
S3002	C001	80
S3001	C002	92
S3003	C003	76

表 5.6　不符合关系模型基本规范的学生表

学号	姓名	基 本 属 性		所 在 院 系
		性别	年龄	
S3001	张以	男	18	计算机学院
S3002	赵丽	女	19	管理科学与工程学院
S3003	李静	女	18	信息学院

在关系模型中，能够唯一标识某个元组的一个属性（或属性集合）称为关系的主键或主码（Primary Key）。唯一标识是指关系中的任何两个元组在该属性（或属性集合）上的值都不相同。例如，学生表中的主键是学号，课程表中的主键是课程号，选课表中的主键为（学号、课程号），这种由多个属性共同组成的主码被称为复合主码。

如果某个属性 A 虽然不是关系 X 的主键（或者只是主键的一部分），但它却是另一个关系 Y 的主键，则称属性 A 为关系 X 的外键或外码（Foreign Key）。例如，表 5.4 课程表中的任课教师为课程表的外键，关联一个新关系：教师表（教师工号、教师姓名、开始工作时间）。关系 X 称为引用关系，关系 Y 称为被引用关系。

将 E-R 图中表示的概念模型转换为关系数据模型要遵循以下原则。

（1）E-R 图中一个实体被转换为一个关系，关系的属性为实体的属性，关系的主键为实体的码。

（2）一个 $m:n$ 的联系可以转换为一个关系，与该联系相关的各个实体的码以及联系本身的属性可以转换为关系的属性，其中，各个实体的码组成关系的主键。

（3）一个 $1:n$ 的联系可以转换为一个关系，也可以与 n 端对应的关系合并。如果转换为一个独立的关系，转换规则与 $m:n$ 联系的转换规则相同。

（4）一个 $1:1$ 的联系可以转换为一个关系，也可以与任意一端对应的关系合并。

在关系数据模型中，为了保障数据的完整性和一致性，需要对关系设置一些约束条件，称为完整性约束。关系数据模型中的完整性约束主要包括实体完整性、参照完整性和用户自定义完整性。

（1）实体完整性。

实体完整性约束是指关系中的主键对应属性的取值不能为空，且取值必须唯一，不能重

复。例如，学生表中的学号不能为空，且不能出现两个学生实体的学号相同的情况。

(2) 参照完整性。

参照完整性约束是指引用关系中的外键对应属性的取值要么是空值，要么是被引用关系中对应属性已经存在的取值。例如，在选课表中的学号的取值必须在学生表中存在。

(3) 用户自定义完整性。

任何关系数据库系统都必须支持实体完整性和参照完整性。除此之外，用户还可以根据具体应用的语义要求，定义一些特殊的约束条件。例如，规定学生表中学生姓名不能取空值；选课表中成绩的取值范围为 0～100。

对关系数据模型中的数据可以基于关系代数和关系演算进行数据操作，操作类型主要包括数据查询、数据插入、数据删除和数据修改。

(1) 数据查询。

数据查询是从关系中选取满足条件的数据，既可以查询满足条件的元组(选择)，也可以查询指定列(投影)，还可以把不同关系中的数据关联在一起查看(连接)。

(2) 数据插入。

数据插入是将一个新的元组添加到现有的关系中，如某个学生新选了一门课程后，可以把(学号，课程号，Null)插入到选课表中，其中成绩为 Null 是因为该门课程还没有成绩。

(3) 数据删除。

数据删除是指将关系中不需要的元组从关系中去掉，如将退学的学生从学生表中删除。

(4) 数据修改。

数据修改是对关系中已有元组的属性值进行修改，如学生考完试后把成绩对应的空值 Null 修改为该学生这门课程的考试成绩。

3. 结构化查询语言

结构化查询语言(SQL)是一个通用的功能强大的关系数据库语言，用于用户进行数据库模式操纵、数据库数据的操作、数据库安全性完整性定义和控制等一系列功能。SQL 是国际标准语言，大多数关系数据库均使用 SQL 作为数据存取语言，并且语法元素和结构具有共通性，为不同数据库系统间的相互操作奠定了基础。

1) SQL 的发展历史

1974 年，IBM 圣何塞实验室的 Boyce 和 Chamberlin 在研制关系数据库管理系统原型系统 System R 的过程中，提出了一套规范语言 Sequel (Structured English Query Language)，并在 1980 年正式称为 SQL。1986 年 10 月，美国国家标准局(American National Standards Institute，ANSI)采用 SQL 作为关系数据库管理系统的标准语言(ANSI X3. 135—1986)。1987 年，国际标准化组织(International Organization for Standardization，ISO)将 SQL 采纳为国际标准。后来每隔一段时间，ISO 都会更新 SQL 标准的版本。

2) SQL 的组成

SQL 将数据库系统的操作分为三个类别，分别是数据定义语言 (Data Definition Language，DDL)，数据操纵语言 (Data Manipulation Language，DML)和数据控制语言 (Data Control Language，DCL)。

(1) 数据定义语言。

数据定义语言用于创建或删除数据库模式，例如对表、视图和索引等数据库对象的创建

和删除。数据定义语言的语句包括动词 CREATE 和 DROP，数据库对象的名词主要为 TABLE、VIEW 和 INDEX。

（2）数据操纵语言。

数据操纵语言可以对数据库中的数据进行查询、插入、删除和修改，分别对应动词 SELECT、INSERT、DELETE 和 UPDATE。

（3）数据控制语言。

数据控制语言包括除 DDL 和 DML 之外的其他语句，如对访问权限的控制、对安全级别的控制、对连接会话的控制等。常用语句包括 GRANT、REVOKE 等，分别表示授予用户访问权限，解除用户访问权限。

3）SQL 示例

这里给出几个简单的 SQL 示例，其他示例请详见数据库专业书籍。所有 SQL 均默认在 MySQL 数据库中运行，在其他数据库可能需要微调才能正常运行。

（1）数据定义语言示例。

下面是数据库表结构的增加、修改和删除使用 SQL 示例。

【例 5-1】 创建一个 Student 表，表中包括学号（sid）、姓名（name）、性别（sex）和所在院系（department）四个属性。

```
CREATE TABLE Student
(
    sid VARCHAR(11),
    name VARCHAR(25),
    sex VARCHAR(11),
    department VARCHAR(50)
);
```

【例 5-2】 修改 Student 表，新增一个属性 age。

```
ALTER TABLE Student ADD COLUMN age INT FIRST;
```

【例 5-3】 删除 Student 表。

```
DROP TABLE Student;
```

（2）数据操纵语言示例。

【例 5-4】 将一个新学生的信息插入到 Student 表。新学生的学号为 S3004，姓名为张庆，所在院系为计算机学院，性别为女，年龄为 18。

```
INSERT Into Student
(sid,nam,sex,department,age)
VALUES('S3004','张庆','女','计算机学院',18);
```

【例 5-5】 修改表 Student 中学号为 S3004 的学生的 age 为 20 岁。

```
UPDATE Student
SET age = 20
WHERE sid = 'S3004';
```

【例 5-6】 查询表 Student 姓名为张庆的学生所在的年龄和所在院系。

```
SELECT age, department from Student where name = '张庆';
```

【例 5-7】 删除表 Student 姓名为张庆的学生。

```
DELETE FROM Student
WHERE name = '张庆';
```

4. 数据库事务

数据库事务是用户定义的一组操作序列。事务具有四个特性，主要包括原子性(Atomicity)、一致性(Consistency)、隔离性(Isolation)和持久性(Durability)，简称为 ACID 特性。事务的 ACID 特性对数据库的恢复有重要作用。

1) 原子性

事务包括的操作都做或者都不做。例如，某用户想从银行账号 A 转账一万元到银行账号 B，需要的操作包括从账号 A 中减去一万元，在账号 B 中增加一万元。两个操作如果不满足原子性，则容易出现错误。

2) 一致性

事务执行的结果要使数据库从一个一致性状态转换到另一个一致性状态。如果事务尚未完成被迫中断，未完成的事务对数据库做的修改已经有一部分写入到物理数据库，会造成数据库处于不一致的情况。

3) 隔离性

并发事务间是相互独立的，不会互相干扰。

4) 持久性

事务一旦提交，其对数据库中数据的改变是永久性的。即使接下来系统发生故障也不会对执行结果有影响。

5. 常见的关系数据库

自关系数据模型被提出以来，出现了众多的关系数据库管理系统。如仅个人简单使用或学习，可以考虑 Access 数据库。市场上也有很多成熟的商用 RDBMS 产品，份额较大的主要有 Oracle、SQL Server 和 DB2。除了商业产品，也有一些关系数据库管理系统是开源的，比较流行的开源产品有 MySQL、PostgreSQL 和 SQLite。

1) Access

Office Access 是 Office 中的一个成员，以一定的格式将数据存储在 Access Jet 的数据库引擎中，是一个结合图形用户界面和软件开发工具的关系数据库管理系统。

2) Oracle

Oracle 数据库是美国 Oracle(甲骨文)公司提供的，使用广泛的 RDBMS 产品。

3) SQL Server

SQL Server 最初由微软、Sybase 和 Ashton-Tate 三家公司共同开发，后来微软公司和 Sybase 分别专注于在 Windows 操作系统和 UNIX 操作系统上的应用。现在提到的 SQL Server 是指微软公司推出的关系数据库管理系统，主要运行在 Windows 操作系统上。其使用方便、可伸缩性好、与相关软件集成程度高。2017 年，微软修正了原 SQL Server 无法运行在类 UNIX 操作系统上的缺陷，SQL Server 2017 已经可以支持 Linux 操作系统。

4）DB2

DB2 由 IBM 公司研发，被认为是最早使用 SQL 的 RDBMS。DB2 大多应用于大型应用系统，具有跨操作系统平台的特点，且具有较好的可伸缩性，既可以支持移动计算，又可以支持大型企业级应用。

5）MySQL

MySQL 是由瑞典 MySQL AB 公司开发的、开源的关系数据库管理系统，目前归属于 Oracle 旗下。其具有体积小、速度快、成本低、开放源码等优点，因此被广泛使用。中小型网站的开发一般都选择 MySQL 作为数据库，因此其流行度一直很高，在 DB-Engines 的流行度排行中稳居第二。

6）PostgreSQL

PostgreSQL 由 2014 年图灵奖得主 Michael Stonebraker 领导创建的 Postgres 发展而来，是可以获得的开放源码中最先进的数据库系统。其提供了多种开发语言接口，包括 C、Java、C++、Python 等。在 DB-Engines 的流行度排行中，PostgreSQL 目前位居 Oracle、MySQL 和 SQL Server 之后的第四位。

7）SQLite

SQLite 是用 C 语言编写的数据库引擎，支持跨操作系统运行。SQLite 适合嵌入式或轻量级应用，其在物联网、移动设备等领域将有非常好的发展机会。

5.3.2　NoSQL 数据库

关系数据库在大数据时代和 Web 2.0 时代暴露出越来越多的缺陷。主要表现为：无法高效管理海量数据、无法满足数据高并发的需求、无法满足高扩展性的需求。关系数据库无法通过添加更多的硬件和计算机节点扩展负载能力。鉴于此，NoSQL 数据库得以快速地发展。典型的 NoSQL 数据库一般可以划分为键值对数据库、列族数据库、文档数据库和图数据库四类。

1. 键值对数据库

键值对数据库中每一个 Key 指向特定的 Value，Value 可以是任意类型的数据，可以通过 Key 进行查询和定位 Value，但不能通过 Value 进行查询和索引。键值对数据库具有很强的可扩展性，当存在大量的写操作时，其性能会比关系数据库好。键值对数据库可以进一步划分为内存数据库和持久化数据库，前者把数据保存在内存中，后者把数据保存在磁盘中。

2. 列族数据库

在列族数据库中，存储数据的基本单位是一个列，包括列名和值。关联紧密的列可以组合在一起形成列族，实现近邻存储。每行中的列和列族的模式和数量都可以不同。

3. 文档数据库

文档数据库中处理的最小单位是文档，可以用不同的标准，如 JSON、XML 和 BSON 等存储文档内容。在存储文档数据之前，不需要对文档定义任何模式。文档数据库通过键定义一个文档，因此可以看成是键值对数据库的一个衍生品，而且文档数据库比键值对数据库具有更高的查询效率，尤其是基于文档内容的索引和查询，这种基于 Value 值进行查询在普通键值对数据库中是无法进行的。

4. 图数据库

图数据库中存储了图中的顶点以及连接顶点的边。图数据库专门用于处理可以用图进行抽象的应用，如推荐系统、社交网络和知识图谱。

在实际应用中，一些公司会同时采用多种不同的数据库，以适应不同的应用场景。例如，电子商务网站可以使用键值对数据库存储“购物篮”这种临时性数据，用关系数据库存储当前的产品和订单信息，而用 MongoDB 这种文档数据库存储大量的历史订单数据。对四类 NoSQL 数据库的对比以及各自数据模型简单描述分别如表 5.7 和图 5.5 所示。

表 5.7　四类 NoSQL 数据库对比

数据库类型	数据模型	优点	缺点	应用场景	相关产品
键值对数据库	以键值对的形式存储数据，主要采用散列表	查找速度快、扩展性好、灵活性好	数据无结构化，事务不支持回滚	会话、配置文件、购物车等	Redis，Memcached
列族数据库	以列族方式存储	查找速度快，容易进行分布式扩展	功能较少	分布式数据存储于管理	BigTable，HBase，HadoopDB
文档数据库	Key-Value 对应的键值对，Value 一般为 JSON、XML 等格式	数据结构灵活，性能好	缺乏统一的查询语法，查询性能不高	处理面向文档的数据	MongoDB，CouchDB
图数据库	图结构	支持图算法	需对整个图做计算，不容易进行分布式	社交网络、推荐系统等	Neo4j，InfoGrid

5.3.3　分布式文件系统

传统的单台主机采用的文件系统无法应对高效存储个体体量大的文件型数据的挑战，而且无法提供足够的处理能力和扩展性应对数据规模的快速增长。2004 年，谷歌提出了一种并行计算模型 MapReduce，用于处理大规模数据。为 MapReduce 提供数据存储支持的是分布式文件系统(Google File System，GFS)，其实现了大体量文件在多台机器上的分布式存储。同年，Doug Cutting 基于 Java 实现了谷歌 MapReduce 系统，被称为 Hadoop，受到了全球学术界和工业界的普遍关注。HDFS（Hadoop Distributed File System）和 MapReduce 是 Hadoop 的核心组成部分，前者是对 GFS 的开源实现，基于网络实现数据的分布式存储；后者负责分布式计算。本小节主要介绍 HDFS，MapReduce 将在 5.4.2 节中进行介绍。用户可以使用 Sqoop 开源工具在 HDFS 和关系数据库之间进行数据转移，将传统关系型数据库，如 MySQL、Oracle 中的数据转移到 HDFS，或将 HDFS 中的数据导出到传统关系数据库。

1. 计算机集群结构

普通的文件系统依赖单个计算机完成文件的存储和处理。分布式文件系统背后依赖的是由多个计算机节点构成的计算机集群，且这些计算机节点可以是由普通廉价的硬件组成的，大大降低了在硬件上的开销。图 5.6 描述了计算机集群的基本架构，集群中包括 n 个机架，每个机架上可以放 8～64 个计算机节点。同一个机架上的计算机节点间通过网络互联，

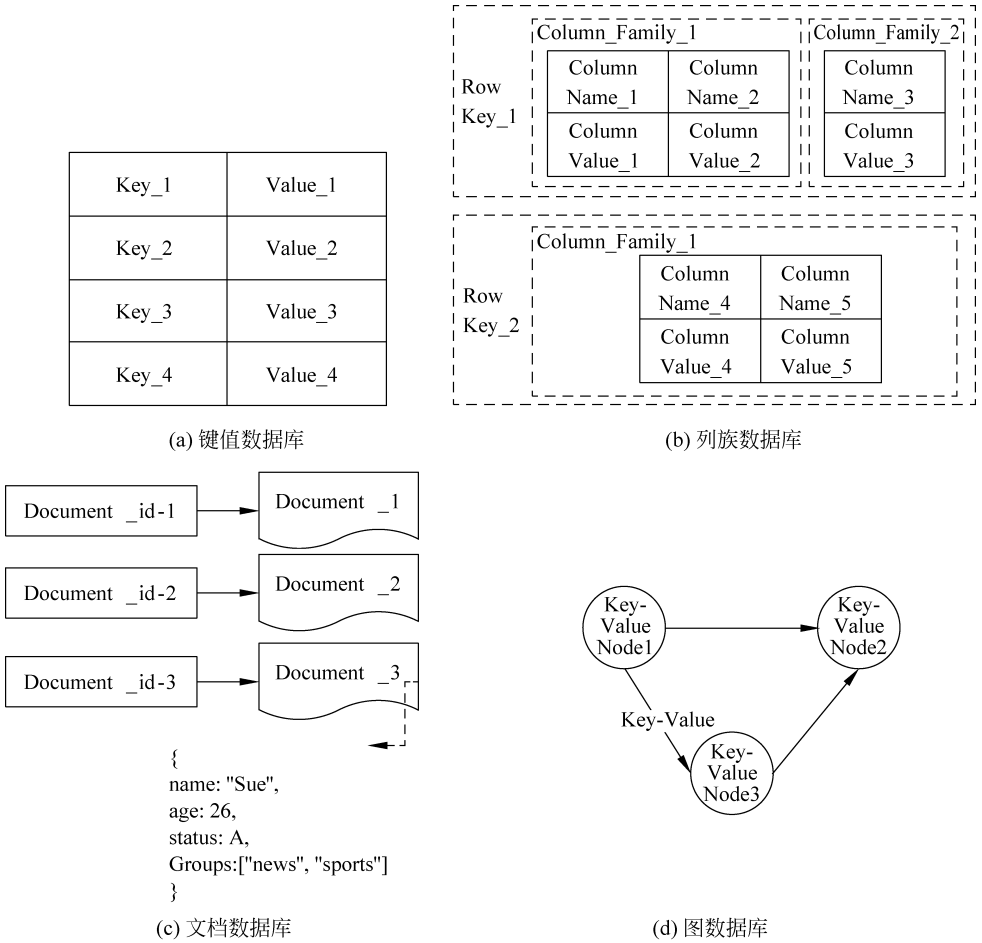


图 5.5　四类 NoSQL 数据库数据模型

不同机架之间一般采用交换机互联。

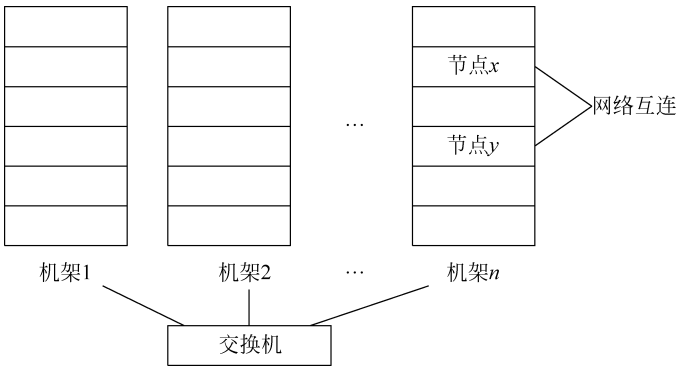


图 5.6　计算机集群的基本架构

2. 分布式文件系统结构

在 Windows 或 Linux 等操作系统中，文件系统会把磁盘空间划分为磁盘块，大小一般为 512B。文件系统中的块(Block)一般是磁盘块的整数倍，即每次读/写的数据量必须是磁

盘块的整数倍。HDFS同样采用了块的概念，不过在块的大小设计上要明显大于普通文件系统，默认的一个块是64MB。这样做的目的是为了最小化寻址开销，以期在处理大规模文件时更有效率。块的大小也不宜设置地过大，这是因为在MapReduce中一次只处理一个块中的数据，如果块太大，会降低作业并行处理速度。

支持分布式文件系统的计算机集群上的节点可以分为两类：一类为“名称节点”(Name Node)，或称为“主节点”(Master Node)；另一类为“数据节点”(Data Node)，或称为“从节点”(Slave Node)。名称节点一方面负责维护文件系统树，存储所有文件和文件夹的元数据，即用来描述数据集本身特征的数据，因此名称节点记录了每个文件中各个块的位置信息；另一方面记录了所有针对文件的操作，如创建、删除和重命名。数据节点负责数据的存储和读取，每个数据节点负责的数据会被保存在各自节点的文件系统中。

在分布式文件系统中，一个文件会被切成若干个数据块，被分布存储在若干个数据节点上。当客户端需要访问某个文件时，需要将文件名发送给名称节点。名称节点会根据文件名找到对应的各个数据块，并将每个数据块在数据节点的位置信息返回给客户端。根据这些位置信息，客户端可以直接访问对应的数据节点读取数据，在此期间，名称节点并不参与数据的传输，而只是起到监督和协调的作用。在存储数据时，由名称节点分配存储位置，随后客户端把数据直接写入数据节点对应的位置。在多用户需要同时对文件进行操作时，名称节点会对正在修改的文件加锁。名称节点会给提交写请求的用户分配租约，只有获得许可才可以进行写操作。在文件写操作执行完成后，用户归还租约，此时其他用户才可以进行读写。通过以上方式，保障了数据的一致性。

由于普通计算机集群中发生硬件故障是种常态，因此HDFS设置了多副本存储机制以保障硬件发生故障后数据的可靠性和完整性。具体来说，在HDFS中，每个文件块会有多份副本(默认为3份，可以由用户设置)。通常不同的副本会储存在不同的计算机节点上，默认的3份副本中有两份放在同一个机架的不同节点上，第三份副本放在不同机架的节点上，这样既可以保证机架发生异常时的数据恢复，又可以提高数据的读写性能。

3. HDFS的特性

HDFS适合“一次写入、多次读取”，比较适合离线批量处理大规模数据。例如，电子商务网站对用户购物习惯的分析，但不适合经常对文件进行更新的在线业务，如股票实盘。HDFS的特性包括以下几个方面。

1) 适合存储和处理大文件

HDFS中的文件一般可以达到吉字节(GB)甚至太字节(TB)级别，目前来看，HDFS的存储和处理能力已经能达到PB级。

2) 兼容廉价的硬件设备

HDFS设计了多种机制，如进行自动恢复、快速硬件故障检测，以保障硬件发生故障时数据的完整性。

3) 采用流式数据读写，而不是随机读写的方式

HDFS为了满足批量数据处理的要求，提高数据吞吐量，放松了一些POSIX的要求，以流式方式访问数据。

4) 强大的跨平台兼容性

HDFS采用Java语言实现，可以运行在任何支持JVM(Java Virtual Machine)的机

器上。

5）HDFS的可伸缩性较强

通过将更多的计算机节点加入进来，集群规模可以横向扩展。

HDFS的局限性主要包括以下几个方面。

1）不适合低延迟的应用

这与HDFS采用流式数据读写的方式有关。而在低延迟的应用场景，往往需要通过数据库访问索引的方式进行随机读写，HBase是更合适的选择。

2）无法高效存储大量小文件

小文件是指文件小于HDFS中block size(默认64MB)的文件。一方面，小文件数量太多，名称节点中文件元数据的存储和查找都会出现瓶颈；另一方面，访问大量小文件会频繁从一个数据节点跳到另一个数据节点，严重影响设备性能。此外，处理大量小文件还会在分布式计算时因产生过多的Map任务而大大增加线程管理开销。

3）不支持多用户并发写入，不支持任意修改文件

对文件执行写操作时，只允许追加操作，不支持随机写操作。

5.4　大数据计算

本节简要介绍大数据计算的分类，包括批量计算、流式计算和大规模图计算，并重点介绍用于批量计算的MapReduce并行计算技术。

5.4.1　概述

根据应用场景和处理对象的特点不同，大数据计算主要包括批量计算、流式计算和大规模图计算。批量计算用于离线计算场景，处理的数据是静态的，在计算过程中数据不会发生变化。例如，对淘宝2019年所有商品的交易记录进行分析，统计年度销量最高的商品。常见的大数据批量计算系统包括分布式并行编程框架MapReduce和基于内存的分布式计算框架Spark等。流式计算主要用于在线计算场景，处理的数据是动态的。例如，实时统计某网站的访客数，当一个新访客到来时，访客数就要实时加1。采用流式计算可以满足这些应用场景对实时性的要求。常见的大数据流式计算系统包括Twitter支持开发的Storm、Spark Streaming、S4(Simple Scalable Streaming System)、Facebook的Data Freeway和Puma等。

5.4.2　MapReduce

传统的计算大多在单台计算机上开展，这种方式使得程序的性能受到单台机器性能的局限，也无法处理大规模数据。而分布式并行编程可以将程序运行在由大量计算机节点组成的计算机集群上，充分利用集群的并行处理能力，且可以通过向计算机集群中增加新的计算机节点的方式不断增强数据处理能力。

谷歌公司最先提出了分布式并行编程模型MapReduce，之后开源项目Hadoop将其实现。MapReduce的计算过程主要包括两个函数：Map和Reduce。在MapReduce中，存储在分布式文件系统中的大规模数据集会被切分为独立的数据块，这些小数据块可以被多个

Map任务并行处理。随后，Map任务产生的结果以<key，value>的形式分发给多个Reduce任务，其中具有相同key的结果会被发送给同一个Reduce任务。Reduce任务对这些<key，value>的中间结果进行汇总合并，将最后结果写入到分布式文件系统中。在计算过程中，不同的Map任务间是完全相互独立的，不会进行通信，不同的Reduce任务间也不会发生信息交换。

以统计文本中所有单词的出现频次直观地展现一下MapReduce的计算过程，如图5.7所示。在本例中，一个文档被切分为3个数据块，每个数据块包含一行文本。每个数据块由一个Map任务处理，因此共有3个Map任务。因Map任务需要以<key，value>的形式作为输入，以文档中文本的行号作为key，以该行的内容作为value。接着，对Map的输出结果进行Shuffle，即进行分区、排序和合并。在Shuffle过程中，还可以支持用户自定义Combiner函数，若用户没有定义Combiner函数，则不用进行合并操作，即不用将具有相同key的value相加。如果定义了Combiner操作，图中以"dogs"为key的<"dogs",<1,1>>将合并为<"dogs",2>。Shuffle的结果将会作为Reduce任务的输入，最终输出文档中每个单词出现的次数。

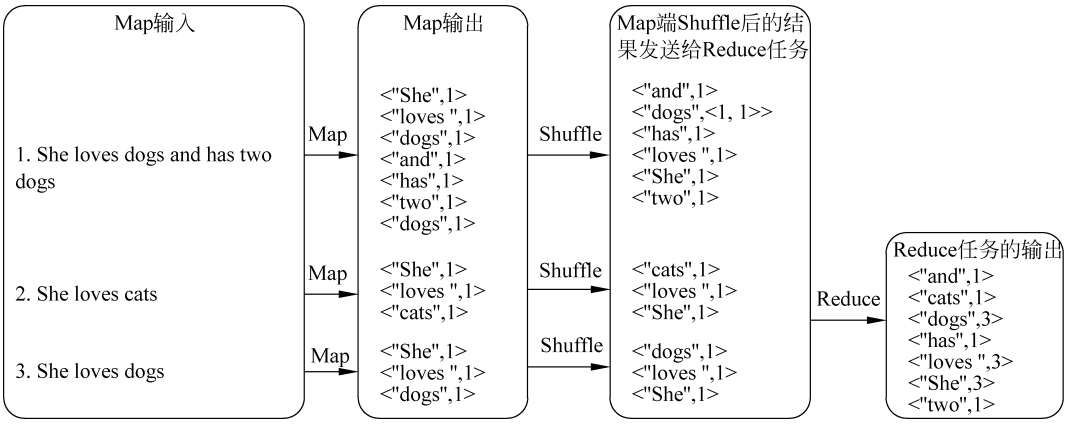


图5.7 MapReduce的计算过程

5.5 数据分析

本节首先划分数据分析的三个层次：描述性分析、预测性分析和规则性分析；介绍描述性数据分析的常用统计指标；梳理总结典型的预测性分析任务，分析机器学习的基本逻辑，介绍数据挖掘和机器学习的常用算法，重点介绍决策树和人工神经网络两种算法，最后列出常用的工具包。

5.5.1 概述

数据分析是将“数据”转换为“知识”，并进一步转化成“智慧”的关键环节。数据分析可以分为三个层次：描述性分析、预测性分析和规则性分析。

描述性分析可以概括数据的位置特征、分散性和关联性等数字特性，并可以反映数据的整体分布特征。用到的统计指标主要包括均值、中位数、众数、标准差、极差等。描述性分析

只关注过去的数据，并不能预测未来。

预测性分析用于预测未来的概率和趋势，在大量历史数据的基础上，建立科学的模型，当新数据到来时，就可以对新数据进行预测。预测性分析采用的技术主要包括数据挖掘和机器学习。数据挖掘(Data Mining)是从大量的、不完全的、有噪声的、模糊的数据中，提取隐含在其中的，人们事先不知道的，潜在有用的信息和知识的过程。机器学习(Machine Learning)对于某给定的任务 T，在合理的性能度量方案 P 的前提下，某计算机程序可以自主学习任务 T 的经验 E，随着提供合适、优质、大量的经验 E，该程序对于任务 T 的性能逐步提高。这里对数据挖掘和机器学习的概念不做严格的区分。

规则性分析是利用仿真和优化方法，旨在给定约束条件下得到最优的解决方案。

5.5.2　数据描述性分析

描述性分析的常见统计指标可以用来测量数据的集中趋势和离散程度。其中，均值、中位数和众数用来描述数据的集中趋势，标准差、方差、极差用来测量数据的离散程度。

1. 均值

均值可以反映数据的平均水平，假设 n 个一维数据分别为 $x_1, x_2, \cdots, x_n$，则均值 $\bar{x}$ 可以表示为 $\bar{x}=\frac{1}{n}\sum_{i=1}^{n}x_i$。例如，数据(80,75,96,78,85)的均值是 $\bar{x}=\frac{80+75+96+78+85}{5}=82.8$。均值容易受极端值影响。

2. 中位数

中位数是数据按照大小顺序排列后，位于中间位置的数。与均值相比，中位数不受极端值影响。假设 n 个一维数据分别为 $x_1, x_2, \cdots, x_n$，则中位数 M 可以表示为

$$M=\begin{cases} x_{\frac{n+1}{2}}, & n \text{ 为奇数} \\ \frac{1}{2}\left(x_{\frac{n}{2}}+x_{1+\frac{n}{2}}\right), & n \text{ 为偶数} \end{cases}$$

例如，(80,75,96,78,85)的中位数是按照大小顺序排序后数据(75,78,80,85,96)中间位置的数，即 80。

3. 众数

众数是数据中出现最多的数(所占比例最大的数)。与均值相比，众数不易受极端值的影响。一组数据中，可能存在多个众数，也可能不存在众数。例如，在 1、2、2、3、3 中，众数是 2 和 3；在 1、2、3、4、5 中没有众数。

4. 方差和标准差

方差和标准差的值越大，数据的离散程度越高，标准差是方差的算术平方根。方差 s^2 与标准差 s 的计算表达式为

$$s^2=\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2$$

$$s=\sqrt{s^2}=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2}$$

其中，n 为数据的个数；$\bar{x}$ 为该组数据的均值。

5. 极差

极差 R 仅关注数据的上下界，被定义为最大值与最小值之差，即 $R=x_{\max}-x_{\min}$。

5.5.3 预测性分析

1. 预测性分析任务

预测性数据分析任务主要包括回归、分类、聚类、关联规则、离群点检测、时间序列预测等。典型的预测性分析任务的描述和示例如表 5.8 所示。

表 5.8 典型的预测性分析任务

任务	任务描述	示例	常见算法
回归	预测结果是数值型数据	① 预测患流感的人数 ② 预测未来的房价	XGBoost、GBDT、随机森林
分类	预测结果是类别型数据	① 预测邮件是垃圾邮件还是非垃圾邮件 ② 预测客户信用风险是高还是低	支持向量机、朴素贝叶斯、决策树、神经网络、逻辑回归、K近邻
聚类	根据样本的相似性将样本划分为不同的簇，使得簇内样本相似度高，不同簇间样本相似度低	客户细分	K-means、DBSCAN
关联规则	发现哪些物品会同时被购买（或同时出现）	啤酒和尿布的例子	Apriori、FP-Growth
离群点检测	识别数据中的离群点，即显著不同于其他数据的数据对象	电信诈骗识别	LOF
时间序列预测	按时间顺序排列形成的数列为时间序列，根据历史时间的数值预测未来某时间（段）的数值	根据零售店历史销售额预测该店未来的销售额	简单移动平均法、移动平均法等、长短期记忆模型 LSTM 等

2. 机器学习

在早期，人们进行预测一般采用基于规则的方法，其逻辑如图 5.8(a)所示。专家需要人工制定系列规则，对于某一条规则，如果满足某种条件，执行 Code 1；如果不满足该条件，执行 Code 2。这些规则需要专家制定，因此需要专业的领域经验，耗时耗力，领域移植性差。

如图 5.8(b)所示，机器学习在训练数据的基础上，应用合适的机器学习算法，会得到相应的模型。模型可以是一个复杂的目标函数，也可以是一系列规则。当需要预测的新样本到来时，就可以应用该模型对新样本进行预测。相对于人工制定规则，机器学习可以从数据中自动地学习到解决问题的方法和规则。

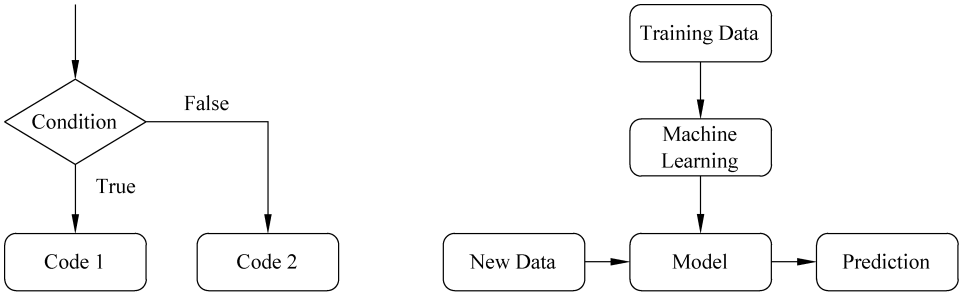
按照机器学习的方式不同，机器学习可以大致分为以下四种。

1) 有监督学习(Supervised Learning)

有监督学习从有标签的数据中建立模型，学习数据和其标签之间的关系，允许对未来数据进行预测。数据标注是有监督学习的必要工作。判断邮件是否为垃圾邮件是有监督学习的典型的例子，在进行机器学习之前，需要事先准备好一些训练数据，该数据主要包括两部分：邮件转换成的特征向量以及人工判断其是否是垃圾邮件的标注 label。

2) 无监督学习(Unsupervised Learning)

无监督学习处理的是不带标签的数据，其目标是从数据中自动地发现模式。典型的应



(a) 基于规则的方法　　(b) 基于机器学习的方法

图 5.8　基于规则的方法和基于机器学习的方法

用包括聚类和离群点检测。

3）半监督学习(Semi Supervised Learning)

半监督学习同时使用带标签数据和不带标签的数据，以应对标签数据难以获得的情况。

4）强化学习(Reinforcement Learning)

强化学习利用无标签数据，通过人工的奖惩信号持续改进性能的一种学习类型。

3. 数据挖掘和机器学习常用算法

国际权威的学术组织 the IEEE International Conference on Data Mining (ICDM)2006 年 12 月从 18 种算法中，评选出了数据挖掘领域的十大经典算法：C4.5、K-Means、SVM、Apriori、EM、PageRank、AdaBoost、KNN、Naive Bayes 和 CART。这些算法在数据挖掘和机器学习领域有重要的地位和影响。因篇幅限制，本节重点介绍两种常用的算法—决策树和神经网络。数据挖掘十大算法中 C4.5 和 CART 都属于决策树算法；选择介绍神经网络是因为目前比较流行的深度学习的基础是神经网络。如果读者对其他算法感兴趣，可以参阅其他专业书籍。

1）决策树

决策树既可以解决分类问题，也可以应用于回归问题。分类决策树模型是一种树形结构，描述了对样本进行分类的过程。利用大量数据训练得到的决策树模型就是一棵如图 5.9 所示的树，所使用的数据集如表 5.9 所示。其中，叶子节点代表的“好瓜”和“坏瓜”是要预测的两个类别，中间节点“脐部＝?”“色泽＝?”“根蒂＝?”和“纹理＝?”代表的是数据集的属性。

2）决策树的学习

决策树的学习包括三个步骤：特征选择、决策树的生成和决策树的剪枝。其中，特征选择和决策树的生成作用于表 5.9 中的训练集上，决策树的剪枝作用在表 5.9 中的验证集上。

(1) 特征选择。

特征选择的主要目的是确定需要以哪个属性作为划分属性，使得划分之后节点的纯度最高。因此，在特征选择阶段需要有选择指标的准则。常用的准则包括信息增益、信息增益比和基尼系数。信息增益是指按照某属性划分之后数据集不确定性下降的程度，而这种不确定性由信息熵来测量。数据集 D 的信息熵可按下列公式进行计算。

$$H(D)=-\sum_{k=1}^{Y}\frac{|C_k|}{|D|}\log_2\frac{|C_k|}{|D|}$$

其中，$|D|$是数据集中样本的个数；Y 是数据集中类别的个数；$|C_k|$代表数据集中属于类别 k 的样本的个数。信息熵的范围为[0,1]，熵越大说明纯度越低，熵越小说明纯度越高。

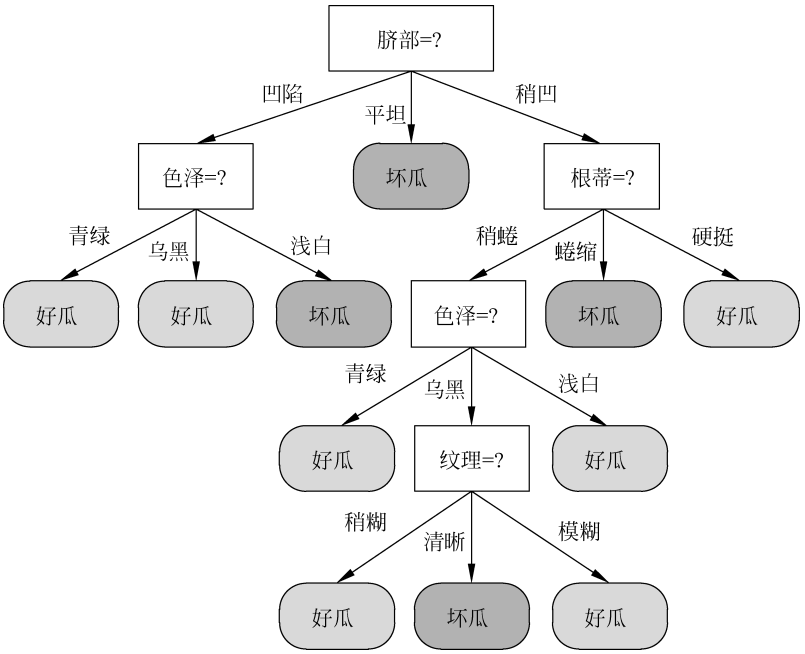


图 5.9　决策树示例

表 5.9　判断是不是好瓜的数据集

数据集	编号	色泽	根蒂	纹理	脐部	好瓜
训练集	1	青绿	蜷缩	清晰	凹陷	是
	2	乌黑	蜷缩	清晰	凹陷	是
	3	乌黑	蜷缩	清晰	凹陷	是
	6	青绿	稍蜷	清晰	稍凹	是
	7	乌黑	稍蜷	稍糊	稍凹	是
	10	青绿	硬挺	清晰	平坦	否
	14	浅白	稍蜷	稍糊	凹陷	否
	15	乌黑	稍蜷	清晰	稍凹	否
	16	浅白	蜷缩	模糊	平坦	否
	17	青绿	蜷缩	稍糊	稍凹	否
验证集	4	青绿	蜷缩	清晰	凹陷	是
	5	浅白	蜷缩	清晰	凹陷	是
	8	乌黑	稍蜷	清晰	稍凹	是
	9	乌黑	稍蜷	稍糊	稍凹	否
	11	浅白	硬挺	模糊	平坦	否
	12	浅白	蜷缩	模糊	平坦	否
	13	青绿	稍蜷	稍糊	凹陷	否

假设某个属性 a 有 v 个取值$\{a_1,a_2,a_3,\cdots,a_v\}$,根据该属性的取值可以将样本集 D 划分为 v 个子集 $D_1,D_2,D_3,\cdots,D_v$。记$|D_i|$为子集 D_i 中样本的个数。则属性 a 在数据集 D 上的信息增益可以计算为

$$IG(D,a)=H(D)-\sum_{i=1}^{v}\frac{|D_i|}{|D|}H(D_i)$$

依次计算每个属性的信息增益，选择信息增益最大的属性作为该次划分的最优属性。

（2）决策树的生成。

决策树的生成算法的框架如图 5.10 所示。

输入：数据集D，数据集中的属性集A

输出：决策树T

决策树生成过程：

(1) 如果D中所有样本都属于同一个类别C_k，则返回单节点树T;

(2) 如果$A=\emptyset$，则T为单节点树，节点的类别为D中样本数最大的类别，返回T;

(3) 否则，计算A中各特征在数据集D上的准则，如信息增益，选择最优的属性a;

(4) 如果属性a的准则满足停止条件，如信息增益小于预先定义的阈值，则置T为单节点树，节点的类别为D中样本数最大的类别，返回T;

(5) 否则，根据属性a的取值将D分割为若干个数据子集D_i，将D_i中样本数最大的类别作为该节点的标记，构建子节点，返回由节点及其子节点构成的树T;

(6) 对节点i，$D\leftarrow D_i$，$A\leftarrow A-a$，重复步骤(1)~(5)。

图 5.10　决策树生成算法

（3）决策树的剪枝。

在决策树生成过程中，学习过程会尽量去适应训练数据，可能会导致决策树分支过多，进而造成在训练集上生成的决策树模型表现比较好，但应用在新样本上的表现却不尽如人意的情况，这种现象称为过拟合。为了避免出现过拟合，需要对决策树进行剪枝。根据剪枝的时机，可以划分为“预剪枝”和“后剪枝”。预剪枝的基本思路是在节点划分之前估计本次节点划分是否会带来泛化能力的提升。泛化能力是指模型对训练集样本以外的新样本的预测能力。在决策树剪枝过程中，用验证集中的样本作为衡量决策树模型泛化能力的“新样本”。如果划分该节点不能带来泛化能力的提升，则不进行此次节点划分。后剪枝是指先产生一颗完整的决策树，然后自下而上将非叶节点作为根节点的子树变为单个节点，看其能否带来泛化性能的提升。如果泛化能力确实提升，则将此子树剪掉，用叶子节点代替。

3）人工神经网络

人工神经网络(Artificial Neural Network，ANN)学习借鉴了生物学的简单理论，其目的是从训练样本中学习到目标函数。神经元是人工神经网络的重要组成部分，因此先来看一下神经元的组成。如图 5.11 所示，神经元由求和函数和激活函数两部分组成，其中激活函数的目的是提升模型的非线性表达能力。常见的激活函数包括 Sigmoid、ReLU、Softmax 等。x_i 表示样本数据的第 i 个特征，w_i 表示对应的权重。

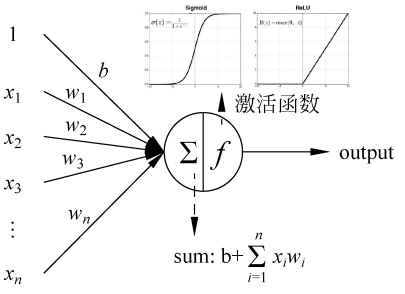


图 5.11　神经元

人工神经网络中多层感知机由输入层、输出层和多个隐含层构成，每一层由若干个神经元组成，如

图 5.12 所示。对于分类任务来说，样本数据被转换成特征向量作为输入，经过多层隐含层，最后输出样本属于每个类别的概率，选择其中概率最大的类别作为该样本的预测结果。神经网络的网络结构，即隐含层的层数以及每层神经元的个数，需要由人提前指定。训练神经网络的目的是确定连接两个神经元的权重的值。要达到这个目的，首先定义损失函数，在分类问题中，经常以输出结果和实际标签的交叉熵作为损失函数，然后随机初始化权重，再利用优化算法（如梯度下降法）沿着使损失函数降低的方向不断调整各权重。当所有权重确定后，将新样本转换为同样维度的特征向量后，就可以通过前馈神经网络计算出每个输出节点的概率。

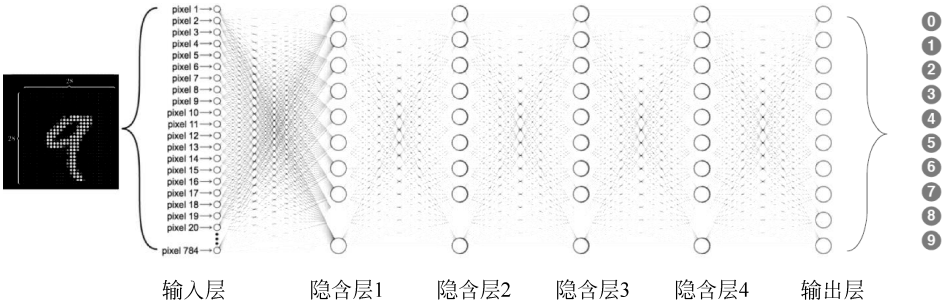


图 5.12　多层感知机

图 5.13 展示了人工神经网络的训练过程。

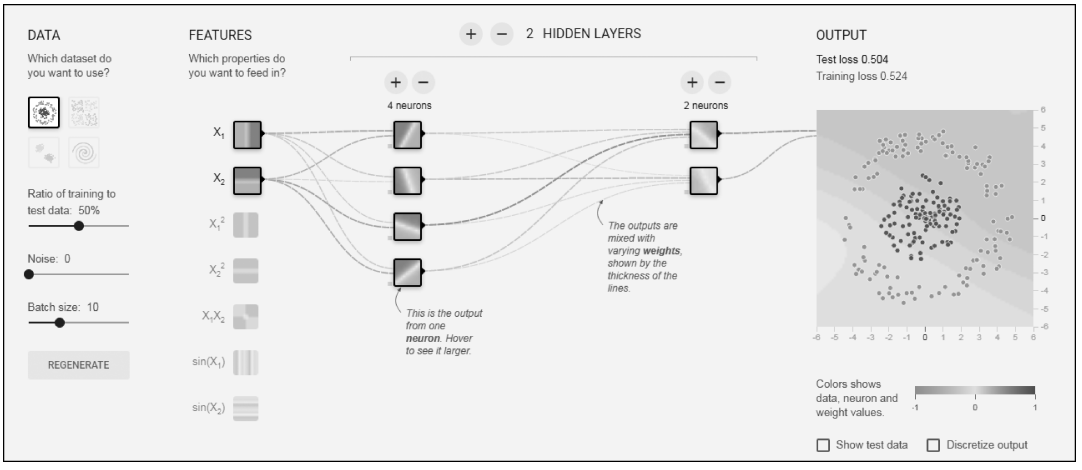


图 5.13　人工神经网络训练过程示例

（图片来源于 http://playground.tensorflow.org）

随着大数据的积累和计算能力的提升，深度学习（Deep Learning）取得了长足的发展。深度学习的概念起源于人工神经网络，是具有多隐含层的网络结构，可以提取抽象的高层特征。2012 年，Hinton 课题组使用深度学习模型 AlexNet 首次参加 ImageNet 图像识别比赛，一举夺得冠军，使得深度学习再次走入人们的视线。2012 年 6 月，《纽约时报》报道了由著名的斯坦福大学的机器学习教授 Andrew Ng 和大规模计算机系统方面的专家 Jeff Dean 教授共同主导的 Google Brain 项目，该项目用 16 000 个 CPU Core 的并行计算平台对 20 000 个不同物体的 1400 万张图片进行辨识，训练出了含有 10 亿个节点的深度神经网络

(Deep Neural Networks, DNN),能够自动识别出猫脸。2014 年 3 月,Facebook 的 DeepFace 项目基于深度学习方法,训练了包含 1.2 亿个参数的 9 层神经网络来获得脸部表征,最终人脸识别技术的识别率达到了 97.25%,几乎可媲美人类。

4. 数据挖掘和机器学习工具

本部分罗列了一些实施数据挖掘和机器学习的工具,包括 Weka、Scikit-Learn,面向深度学习的 Tensorflow、Keras 和 PyTorch,以及运行在并行计算大数据平台上的 Mahout 和 Spark MLib。感兴趣的读者可以前往相关网站自行查看。

5.6　数据可视化

本节首先梳理可视化的发展,介绍数据可视化的作用和几个数据可视化典型案例,最后总结可用来进行数据可视化的工具和软件。

5.6.1　概述

可视化(Visualization)可以将数据转换成图形图像并提供交互,从而帮助用户更加有效地完成数据分析与数据理解等任务。可视化技术在很早就被用来帮助人们展示、分析和理解数据。本部分给出了几个数据可视化的例子。

1. 网络数据可视化

网络中的节点一般表示的是现实世界中的实体,网络中的边通常代表实体间的联系。通过网络数据可视化,可以直观地展现网络中实体的聚集情况。现实世界中,网络数据可以用于分析和展示微博、微信等社交网站中的好友关系,不同学者发表论文的合作关系等。

图 5.14 展示了某位用户 Ali Imam 的好友关系图。从图中可以清楚地看到他的好友划分到了三个主要的群体,并用不同的颜色表示。其中,蓝色节点表示的是 Linkin 中他的好友群,橙色代表的是他的卡内基·梅隆大学的同学圈,青绿色和紫红色节点表示的是他在 Yahoo 工作时的同事圈,其中,紫红色节点代表的是他在 Yahoo Analytics 的同事,而青绿色节点是 Yahoo 其他部门中的同事。从图中可以发现某些有趣的信息,例如,可以发现有几个节点位于两个群体间,代表不同好友圈中的桥梁。

图 5.15 是由 Ramio Gómez 绘制的编程语言间的影响力关系图。该图是一个由不同的编程语言(节点)以及它们间的影响关系(边)建立的有向图。值得注意的是,图中节点的大小表示了该语言的影响力的大小。

2. 吉米·亨德里克斯的音乐播放情况图

基于吉米·亨德里克斯在 1967—1970 年的现场表演数据,绘制了图 5.16,直观地展示了他的歌曲以及在 YouTube 上的播放数据情况。

5.6.2　数据可视化工具和软件

目前有很多数据可视化的工具和软件,可以实现多种数据可视化功能。其中有些不需要编程基础,通过拖曳即可实现,大大降低了软件的使用门槛;也有一些工具需要编写程序,适合对相关编程工具有初步了解的人员。

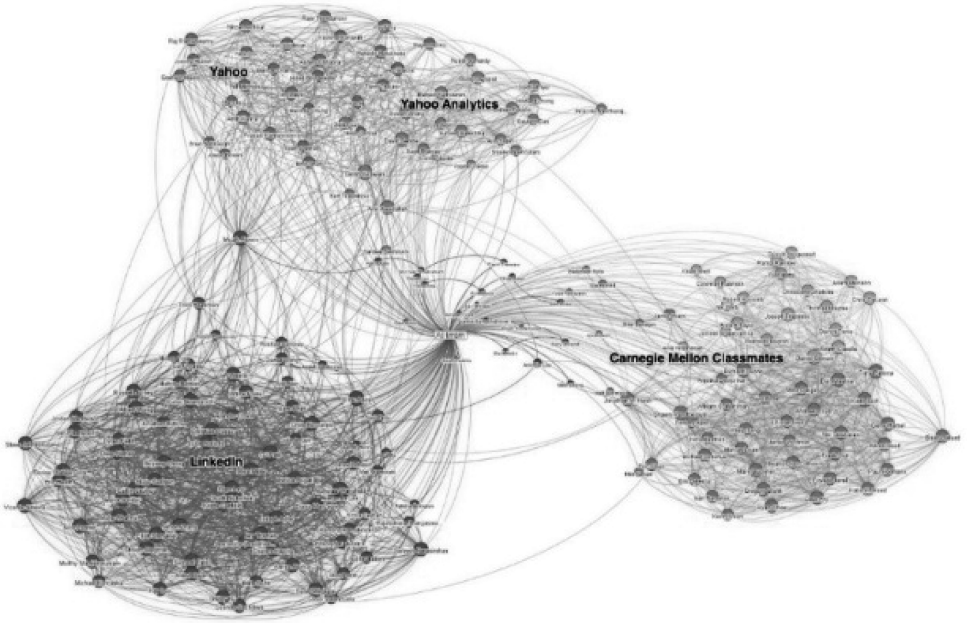


图 5.14　某用户的社交网络

（图片来源于 https://blog.linkedin.com/2011/01/24/linkedin-inmaps）

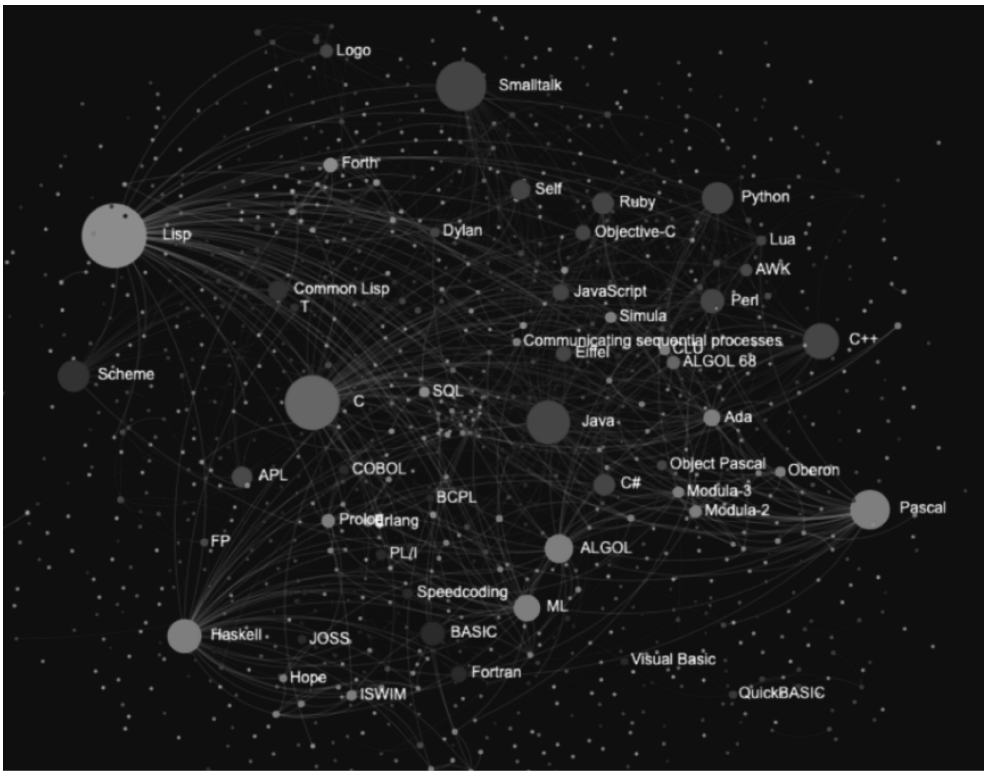


图 5.15　编程语言间的影响关系

（图片来源于 https://exploring-data.com/vis/programming-languages-influence-network/）

图 5.16　吉米·亨德里克斯的音乐播放情况图

（图片来源于 https://public.tableau.com/zh-cn/gallery/jimi-hendrix-live?gallery=votd）

1. 不需要编程的工具

1）Excel

作为常用办公软件 Office 的系列软件之一，Excel 是普通用户进行数据可视化的首选工具，其提供了丰富的图表功能，如柱状图、折线图、饼状图，可以满足日常需求。Excel 简单易学，使用门槛较低。

2）Tableau

Tableau 能够帮助人们查看并理解数据，主要包括 Tableau Desktop、Tableau Prep、Tableau Server 等产品。除了强大的数据可视化功能，Tableau 更是一款集成的 BI 分析软件，主要表现在其强大的数据连接、数据刷新、数据准备和处理功能。Tableau Desktop 支持多种数据源，包括各类常用数据库、json 文件和 Salesforce（销售报表）等，可以实现报表定时更新。Tableau Prep 使分析人员可以更直观、直接地合并、调整和清理数据，以进行下一步的数据分析与数据可视化。Tableau 需要用户进行相应的学习，使用门槛相对 Excel 较高。

3）大数据魔镜

大数据魔镜为用户提供了直观的拖曳界面，帮助用户生成交互式图表，并可以整合多种数据源，包括 MySQL 数据库、ERP 数据、社会化数据等。大数据魔镜已经被广泛应用于电商、金融、互联网、食品、通信、能源、教育等领域。它提供了 500 种可视化效果，并提供了仪表盘功能。魔镜仪表盘支持拖曳式自由布局，提供了丰富的图文组件和多种配色方案。

4）BDP 个人版

BDP 个人版使分析人员通过简单的拖曳就可轻松完成数据整合、数据处理和数据可视化分析。BDP 提供了几十种可视化图表。

5）Gephi

Gephi 是一款开源免费的、公认的网络数据可视化与分析软件之一。Gephi 能够轻松

处理十万个节点的大规模网络数据，也可以计算度数、中心性等常见指标。

Excel、Tableau、大数据魔镜和 BDP 大多处理的是结构化数据，对于文本数据，本部分将介绍几种标签云生成工具，包括 Wordle、Worditout、Tocloud 和微词云，图 5.17 展示了一个词云图。

图 5.17　词云图

6）Wordle

Wordle 可以利用文本或网站词频生成标签云，并提供多种展示风格，允许用户选择文字字体或自定义颜色，生成的标签云可供查看与下载。Wordle 目前只支持英文。

7）WordItOut

WordItOut 操作简单，并提供多种展示风格。用户可以根据需要进行颜色、字符、字体、背景、文字位置等内容的再设计，生成的标签云可供查看与下载。WordItOut 不能识别中文，如果输入中英混合的文本，只能显示英文字体。

8）ToCloud

ToCloud 是一款国产的、免费的标签云生成器，支持中英文。它能提取短语，支持用户设置词的长度和频率。

9）微词云

微词云是一款实用简单的国产在线词云生成器，界面友好、支持中文。用户可以自定义形状、字体和元素等，并支持效果优化。

2. 可视化编程工具

对于有一些编程基础的用户，也可以考虑一些可视化编程工具。

1）ECharts

ECharts 是百度开发的，使用 JavaScript 实现的开源可视化库，可以生成互动图形，并可以在移动设备上和 PC 端流畅运行，能够兼容当前多种浏览器。ECharts 提供了多种常

规的可视化图形，如折线图、饼图、柱状图、散点图、K 线图、盒形图，也提供了酷炫的地理坐标/地图、热力图、关系图、仪表盘等，并且支持不同图形间的混搭。为了和 Python 进行对接，即使用 Python 生成 Echarts 图表，可以使用 pyecharts 包。

2）D3

D3(Data-Driven Documents)是目前比较流行的可视化库之一，使用 JavaScript 实现，具有丰富的 API，可以生成多种互动图形，并支持网页展示。D3 不仅提供了常规图形，还提供了多种复杂的可视化图表形式，如树状图、词云图和圆形集群图。

3）Matplotlib

Matplotlib 是由 Python 实现的，功能完善的绘图库。开发者可以仅通过几行代码生成直方图、条形图、散点图等常规图形以及三维图、等高线图等。

4）ggplot2

ggplot2 是 Hadley Wickham 使用 R 语言编写的绘图包，能够用简洁的函数构建各类图形。它将图形视为由从数据到几何对象(如点、线)和图形属性(如颜色、形状、透明度)的一个映射，通过定义各种底层组件(如方块、线条)来合成图形。

除了上述可视化工具和软件外，还有一些其他的工具如 Google Chart API、Visual. ly，以及专门生成时间线的 Timetoast 和 Xtimeline，专门生成地图的 Modest Maps 和 Leaflet 等。

5.7　本章小结

本章对大数据相关概念、特征、应用等方面进行了概述，梳理了数据流程处理框架，然后依次对数据处理流程中的各环节进行了介绍，包括数据采集与处理、数据存储、大数据计算、数据分析和数据可视化。

习　　题

一、判断题

1. 结构化数据先有结构，后有数据。（　　）

2. 相对于因果关系，大数据更关注相关关系。（　　）

3. 数据可视化的主要目的是为了美观。（　　）

4. 在描述用户评论产品的 E-R 图中，用户实体和产品实体之间是多对多的联系。（　　）

5. HDFS 更适合处理大量小文件。（　　）

6. 中位数和均值都不易受极端值影响。（　　）

7. 在决策树算法中进行选择属性进行划分时，可以选择所有属性中信息增益最小的属性。（　　）

二、选择题

1. 买饮料时，选择的大杯、中杯、小杯属于(　　)。

A. 类别型数据　　B. 序数型数据　　C. 数值型数据　　D. 以上都不是

2. 智能健康手环，是(　　)数据采集技术的应用。

A. Flume　　B. 网络爬虫　　C. API　　D. 感知设备

3. 数据清洗不包括(　　)。

A. 去除重复数据

B. 补全缺失值

C. 发现和识别异常值

D. 将不同量纲的数据缩放到特定的数据范围

4. 为了体现学生和一卡通的对照关系，可以(　　)(假设一卡通即使补办其一卡通编号也与原编号相同)。

A. 在学生表中加上一卡通编号字段

B. 在一卡通表中加上学号字段

C. 新建一张包含一卡通编号和学生编号的表

D. 以上都可以

5. 下列数据分析任务属于回归问题的是(　　)。

A. 识别电信诈骗

B. 根据店铺历史销售额预测未来销售额

C. 预测明天是否会下雨

D. 根据房屋的面积、位置、户型等预测房价

三、填空题

1. 生产数据的方式经历了运营式系统阶段、用户原创内容阶段与_____。

2. 大数据的“4V”特征是指_____、_____、_____、_____。

3. 数据预处理的主要工作包括_____、_____和数据变换。

4. 关系数据库中的实体完整性约束要靠关系的_____来保障；参照完整性约束主要关注的是关系的_____。

5. 支持分布式文件系统的计算机集群上的节点可以分为两类：_____和_____。

6. 人工神经网络中，神经元主要包括求和函数和_____。

四、简答题

1. 简述大数据的含义。

2. 简述数据处理的流程以及每个环节的主要工作。

3. 在你的专业领域，可收集到的数据都有哪些？分别可以采用什么方法进行数据采集？