信息科学技术学术著作丛书

面向数据挖掘的算法设计与分析

翟俊海　张素芳　著

科学出版社

信息科学技术学术著作丛书

面向数据挖掘的算法设计与分析

翟俊海　张素芳　著

科学出版社

北京

内 容 简 介

本书以数据挖掘为应用载体，按应用频率的高低，系统地介绍分治算法、贪心算法、搜索算法和动态规划算法. 同时，介绍算法分析所用的渐近符号及常用的分析方法，包括递归分析方法、非递归分析方法. 本书的特点是结合作者及其团队研究的数据挖掘问题，注重介绍算法的基本思想及算法应用的启发性.

本书可作为计算机科学与技术、软件工程、信息与计算科学、应用数学、自动化等专业本科生和研究生的教材，也可供从事相关研究的科研人员参考.

图书在版编目(CIP)数据

面向数据挖掘的算法设计与分析/翟俊海，张素芳著. —北京：科学出版社，2018.3

(信息科学技术学术著作丛书)

ISBN 978-7-03-056686-7

Ⅰ. ①面… Ⅱ. ①翟… ②张… Ⅲ. ①数据采集-算法设计②数据采集-算法分析 Ⅳ. ①TP274②TP301.6

中国版本图书馆 CIP 数据核字(2018) 第 042296 号

责任编辑：魏英杰 / 责任校对：桂伟利

责任印制：师艳茹 / 封面设计：陈 敬

科学出版社 出版

北京东黄城根北街 16 号

邮政编码：100717

http://www.sciencep.com

艺堂印刷(天津)有限公司印刷

科学出版社发行 各地新华书店经销

*

2018 年 3 月第 一 版 开本：B5(720 × 1000)

2018 年 3 月第一次印刷 印张：12 1/2

字数：249 000

定价：90.00 元

(如有印装质量问题，我社负责调换)

《信息科学技术学术著作丛书》序

21 世纪是信息科学技术发生深刻变革的时代，一场以网络科学、高性能计算和仿真、智能科学、计算思维为特征的信息科学革命正在兴起. 信息科学技术正在逐步融入各个应用领域并与生物、纳米、认知等交织在一起，悄然改变着我们的生活方式. 信息科学技术已经成为人类社会进步过程中发展最快、交叉渗透性最强、应用面最广的关键技术.

如何进一步推动我国信息科学技术的研究与发展；如何将信息技术发展的新理论、新方法与研究成果转化为社会发展的新动力；如何抓住信息技术深刻发展变革的机遇，提升我国自主创新和可持续发展的能力？这些问题的解答都离不开我国科技工作者和工程技术人员的求索和艰辛付出. 为这些科技工作者和工程技术人员提供一个良好的出版环境和平台，将这些科技成就迅速转化为智力成果，将对我国信息科学技术的发展起到重要的推动作用.

《信息科学技术学术著作丛书》是科学出版社在广泛征求专家意见的基础上，经过长期考察、反复论证之后组织出版的. 这套丛书旨在传播网络科学和未来网络技术，微电子、光电子和量子信息技术、超级计算机、软件和信息存储技术，数据知识化和基于知识处理的未来信息服务业，低成本信息化和用信息技术提升传统产业，智能与认知科学、生物信息学、社会信息学等前沿交叉科学，信息科学基础理论，信息安全等几个未来信息科学技术重点发展领域的优秀科研成果. 丛书力争起点高、内容新、导向性强，具有一定的原创性;体现出科学出版社“高层次、高质量、高水平”的特色和“严肃、严密、严格”的优良作风.

希望这套丛书的出版，能为我国信息科学技术的发展、创新和突破带来一些启迪和帮助. 同时，欢迎广大读者提出好的建议，以促进和完善丛书的出版工作.

中国工程院院士

原中国科学院计算技术研究所所长

前　言

在计算机科学中, 通俗地讲, 算法是以计算机作为工具求解问题的方法步骤. 为求解具体问题而设计的算法, 最终要用某种程序设计语言 (如 C/C++ 语言、JAVA 语言、MATLAB 语言等) 编程实现, 然后到计算机上去运行, 以得到所需要的结果。显而易见, 算法在计算机科学中处于基础而又重要的地位, 没有算法就谈不上后续的程序实现及运行.

虽然已经有不少关于算法设计与分析的著作, 但这些著作基本都不是针对具体的应用撰写的, 其优点是通用性, 但缺点也是明显的, 即算法设计与分析和具体应用相脱离, 算法毕竟是为解决具体问题而设计的. 本书正是针对这一问题, 以数据挖掘应用为切入点, 结合作者及其团队多年来关于数据挖掘与算法设计的教学心得, 以及积累的研究成果撰写的. 本书以数据挖掘中常用的算法设计策略为主线, 介绍这些算法设计策略的内容、方法步骤及如何分析算法. 各章节都渗透了这样的思想: "针对具体的数据挖掘任务, 如何设计算法, 如何评估所设计的算法".

本书第 1 章介绍后续章节会用到的预备知识, 包括数据挖掘概述、算法与算法描述的基本概念、算法分析的基本方法等. 第 2 章介绍分治算法. 在这一章, 首先介绍分治算法的基本思想, 给出分治策略的方法步骤. 接下来, 第 1 节介绍归并排序, 第 2 节介绍线性时间选择, 第 3 节介绍基于分治策略的交叉样例选择, 第 4 节介绍大数据 K-近邻算法, 第 5 节介绍大数据样例选择, 第 6 节介绍基于随机上采样和分治策略的两类非平衡大数据分类. 第 3 章介绍贪心算法. 首先通过一个例子, 介绍贪心算法的基本思想和用贪心算法求解问题的步骤, 然后介绍求解背包问题、活动安排问题和旅行商问题的贪心算法. 第 5 节介绍基于贪心策略的特征选择问题, 第 6 节介绍决策树归纳算法, 第 7 节介绍数据挖掘领域应用广泛的梯度下降算法, 第 8 节介绍基于贪心策略的 ELM 网络结构选择问题. 第 4 章介绍搜索算法. 首先介绍两种群体搜索算法: 遗传算法和粒子群优化算法, 然后简单介绍两种个体策略策略: 回溯法和分支限界法. 第 5 章介绍动态规划算法. 这一章首先以多段图问题为例, 介绍动态规划算法的基本思想, 然后归纳出动态规划算法的方法步骤, 接下来介绍求解矩阵连乘问题、0-1 背包问题所有点对之间最短距离问题的动态规划算法.

感谢许宏雨、万丽艳、李塔、邵庆言、苗青、王陈希、庞晓鹤、侯少星、刘博、臧立光、张明阳、王婷婷、郝璞、沈矗和王聪等同学对本书作出的贡献. 本书得到河北大学 "计算机应用技术" 省级重点学科的资助, 也得到了河北省机器学习与计算

智能重点实验室的资助. 本书的出版可为计算机应用省级重点学科从人才培养 (包括青年教师的培养、研究生培养)、科学研究、本科生和研究生教学等方面提供支撑, 为本学科的发展做出贡献. 最后, 感谢科学出版社魏英杰的帮助.

由于水平所限, 书中的不妥之处在所难免, 敬请各位同仁批评指正.

作　者

2017 年 10 月

目　录

第1章　预备知识

本章介绍后续章节将要用到的基础知识, 包括数据挖掘概述、算法与算法描术、算法分析三部分内容.

1.1　数据挖掘概述

数据挖掘 [1-4] 就是从数据中挖掘有应用价值或有潜在应用价值的规律或规则 (也称为知识) 的过程. 需要挖掘的数据具有多种类型, 可能是有结构的数据, 如组织成表结构的数据; 也可能是无结构的数据, 如文本数据; 可能是半结构化的数据, 如 Web 页面数据; 也可能是图像或视频等多媒体数据. 近几年, 随着数据存储技术、网络技术和无线传感技术等的快速发展, 需要处理或挖掘的数据呈现出海量性 (volume)、多样性 (variety)、时效性 (velocity)、准确性 (veracity) 和价值性 (value) 等特性, 满足这 5 种特性 (5V 特性) 的数据称为大数据 [5,6]. 足够多可利用的数据, 可使数据挖掘的质量得到保障. 由于数据挖掘能发现隐藏在数据中的有用信息, 可为企业 (特别是对互联网企业、电商企业、金融企业等) 带来显著的经济效益, 这使得数据挖掘成为非常热门的研究领域, 其应用越来越广泛. 近几年, 针对大数据的数据挖掘研究已经成为学术界和企业界广泛关注的热门话题. 根据要挖掘的知识类型, 数据挖掘的任务可分为分类与回归、聚类分析、关联分析、时间序列分析和偏差检测.

1.1.1　分类与回归

为了易于理解, 同时便于描述, 假设数据挖掘的对象是组织成表结构的数据. 如果数据表中包含样例的类别信息, 则称这种数据表为决策表, 否则称为信息表. 下面先给出决策表的两种形式化定义, 然后再给出分类问题的定义.

定义 1.1.1　一个决策表是一个二元组 $DT=\{(\boldsymbol{x}_i,y_i)|\boldsymbol{x}_i\in U,y_i\in C,1\leqslant i\leqslant n\}$. 其中, $\boldsymbol{x}_i$ 表示决策表中的第 i 个样例, y_i 表示样例 $\boldsymbol{x}_i$ 所对应的类别标号, U 是决策表中 n 个样例的集合, C 是样例所属类别的集合.

定义 1.1.2　一个决策表是一个四元组 $DT=(U,A\bigcup C,V,f)$. 其中, $U=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n\}$ 是 n 个样例的集合, $A=\{a_1,a_2,\cdots,a_d\}$ 是 d 个描述对象 (或样例) 的条件属性 (或特征) 集合, C 是决策属性 (或类别属性), $V=V_1\times V_2\times\cdots\times V_d$ 是 d 个属性值域的笛卡儿积, V_i 是属性 a_i 的值域, $i=1,2,\cdots,d$, f 是信息函数 :

$U \times A \to V$.

决策表的这两种形式化定义实际上是等价的, 在本书中, 我们会交替使用这两种定义, 包含 n 个样例的决策表的直观表示如表 1.1 所示. 下面给出分类问题的定义.

表 1.1　包含 n 个样例的决策表

$\boldsymbol{x}$	a_1	a_2	$\cdots$	a_d	y
$\boldsymbol{x}_1$	x_{11}	x_{12}	$\cdots$	x_{1d}	y_1
$\boldsymbol{x}_2$	x_{21}	x_{22}	$\cdots$	x_{2d}	y_2
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
$\boldsymbol{x}_n$	x_{n1}	x_{n2}	$\cdots$	x_{nd}	y_n

定义 1.1.3　给定决策表 $DT = \{(\boldsymbol{x}_i, y_i) | \boldsymbol{x}_i \in U, y_i \in C, 1 \leqslant i \leqslant n\}$, 如果存在一个映射 $f : U \to C$, 使得对于任意的 $\boldsymbol{x}_i \in U$, 都有 $y_i = f(\boldsymbol{x}_i)$ 成立. 根据给定的决策表 DT 寻找函数 $y = f(\boldsymbol{x})$ 的问题, 称为分类问题, 函数 $y = f(\boldsymbol{x})$ 也称为分类函数.

说明:

① 在分类问题中, 因变量 y 的取值范围是一个由有限个离散值构成的集合 C, 它相当于高级程序设计语言 (如 C++ 语言) 中的枚举类型. 若 C 变为实数集 $\mathbf{R}$ 或 $\mathbf{R}$ 中的一个区间 $[a, b]$, 则这类问题称为回归问题. 显然, 分类问题是回归问题的特殊情况.

② 函数 $y = f(\boldsymbol{x})$ 不一定有解析表达式, 可以用其他的形式, 如树、图或网络来表示.

③ 如果所有的 V_i 都是实数集 $\mathbf{R}$, 此时 $V = \mathbf{R}^d$.

④ 在数据挖掘中, 分类与回归任务就是从给定的数据集中挖掘 (或发现) 分类函数 (或回归函数)$y = f(\boldsymbol{x})$. 因为在求解分类问题或回归问题时, 要用到样例的类别信息, 所以学习分类函数或回归函数的过程属于有导师学习.

下面举几个分类问题的例子.

例 1.1.1　天气分类问题　天气分类问题 [7] 是一个两类分类问题, 用来预测什么样的天气条件适宜打网球. 天气数据集是机器学习领域中的一个经典数据集, 是一个包含 14 个样例的决策表, 如表 1.2 所示.

天气分类问题数据集有 14 个样例, 即 $U = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_{14}\}$; 4 个条件属性, 即 $A = \{a_1, a_2, a_3, a_4\}$, 其中, a_1 =Outlook, a_2=Temperature, a_3=Humidity, a_4 =Wind, 它们都是离散值属性, 相当于高级程序设计语言中的枚举类型属性. $V = V_1 \times V_2 \times V_3 \times V_4$, $V_1 = \{\text{Sunny, Cloudy, Rain}\}$, $V_2 = \{\text{Hot, Mild, Cool}\}$, $V_3 = \{\text{High, Normal}\}$, $V_4 = \{\text{Strong, Weak}\}$. 决策属性 $C = \{y\}$, y =PlayTennis, 它只取 Yes 和 No 两个

值, 因此天气分类问题是一个两类分类问题. 显然, 从该数据集中找到的分类函数 $y = f(\boldsymbol{x})$ 不可能有解析表达式. 在第 3 章, 我们将看到 $y = f(\boldsymbol{x})$ 可用一棵树来表示.

表 1.2 天气分类问题数据集

$\boldsymbol{x}$	Outlook	Temperature	Humidity	Wind	y(PlayTennis)
$\boldsymbol{x}_1$	Sunny	Hot	High	Weak	No
$\boldsymbol{x}_2$	Sunny	Hot	High	Strong	No
$\boldsymbol{x}_3$	Cloudy	Hot	High	Weak	Yes
$\boldsymbol{x}_4$	Rain	Mild	High	Weak	Yes
$\boldsymbol{x}_5$	Rain	Cool	Normal	Weak	Yes
$\boldsymbol{x}_6$	Rain	Cool	Normal	Strong	No
$\boldsymbol{x}_7$	Cloudy	Cool	Normal	Strong	Yes
$\boldsymbol{x}_8$	Sunny	Mild	High	Weak	No
$\boldsymbol{x}_9$	Sunny	Cool	Normal	Weak	Yes
$\boldsymbol{x}_{10}$	Rain	Mild	Normal	Weak	Yes
$\boldsymbol{x}_{11}$	Sunny	Mild	Normal	Strong	Yes
$\boldsymbol{x}_{12}$	Cloudy	Mild	High	Strong	Yes
$\boldsymbol{x}_{13}$	Cloudy	Hot	Normal	Weak	Yes
$\boldsymbol{x}_{14}$	Rain	Mild	High	Strong	No

例 1.1.2 鸢尾花分类问题 鸢尾花分类问题[2]是一个三类分类问题, 根据花萼长 (Sepal length)、花萼宽 (Sepal width)、花瓣长 (Petal length) 和花瓣宽 (Petal width) 四个条件属性对鸢尾花进行分类. 鸢尾花数据集 Iris 包含三类 150 个样例, 每类 50 个样例, 如表 1.3 所示.

表 1.3 鸢尾花分类问题数据集

$\boldsymbol{x}$	a_1	a_2	a_3	a_4	y
$\boldsymbol{x}_1$	5.1	3.5	1.4	0.2	Iris-setosa
$\boldsymbol{x}_2$	4.9	3.0	1.4	0.2	Iris-setosa
⋮	⋮	⋮	⋮	⋮	⋮
$\boldsymbol{x}_{50}$	5.0	3.3	1.4	0.2	Iris-setosa
$\boldsymbol{x}_{51}$	7.0	3.2	4.7	1.4	Iris-versicolor
$\boldsymbol{x}_{52}$	6.4	3.2	4.5	1.5	Iris-versicolor
⋮	⋮	⋮	⋮	⋮	⋮
$\boldsymbol{x}_{100}$	5.7	2.8	4.1	1.3	Iris-versicolor
$\boldsymbol{x}_{101}$	6.3	3.3	6.0	2.5	Iris-virginica
$\boldsymbol{x}_{102}$	5.8	2.7	5.1	1.9	Iris-virginica
⋮	⋮	⋮	⋮	⋮	⋮
$\boldsymbol{x}_{150}$	5.9	3.0	5.1	1.8	Iris-virginica

Iris 数据集有 150 个样例, 即 $U=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_{150}\}$; 4 个条件属性, 即 $A=\{a_1,a_2,a_3,a_4\}$, 其中, a_1 =Sepal length, a_2=Sepal width, a_3=Petal length, a_4=Petal width, 它们都是连续值属性. $V=V_1\times V_2\times V_3\times V_4$, $V_1=V_2=V_3=V_4=R$, 即 $V=\mathbf{R}^4$. 决策属性 $C=\{y\}$, $y\in\{$Iris-setosa, Iris-versicolor, Iris-virginica$\}$. 由于 Iris 数据集中四个条件属性都是连续值属性, 因此该数据集是一个连续值数据集.

例 1.1.3　助教评估分类问题　助教评估分类问题 [2] 也是一个三类分类问题, 它根据母语是否是英语 (A native English speaker)、课程讲师 (Course instructor)、课程 (Course)、是否正常学期 (A regular semester) 和班级规模 (Class size) 五个条件属性对助教评估分类. 助教评估分类数据集 (teaching assistant evaluation, TAE) 包含三类 151 个样例, 第一类 (Low)49 个样例, 第二类 (Medium)50 个样例, 第三类 (High)52 个样例, 如表 1.4 所示.

表 1.4　助教评估分类问题数据集

$\boldsymbol{x}$	a_1	a_2	a_3	a_4	a_5	y
$\boldsymbol{x}_1$	2	21	2	2	42	Low
$\boldsymbol{x}_2$	2	22	3	2	28	Low
⋮	⋮	⋮	⋮	⋮	⋮	⋮
$\boldsymbol{x}_{49}$	2	2	10	2	27	Low
$\boldsymbol{x}_{50}$	2	6	17	2	42	Medium
$\boldsymbol{x}_{51}$	2	6	17	2	43	Medium
⋮	⋮	⋮	⋮	⋮	⋮	⋮
$\boldsymbol{x}_{99}$	2	22	1	2	42	Medium
$\boldsymbol{x}_{100}$	1	23	3	1	19	High
$\boldsymbol{x}_{101}$	2	15	3	1	17	High
⋮	⋮	⋮	⋮	⋮	⋮	⋮
$\boldsymbol{x}_{151}$	2	20	2	2	45	High

TAE 数据集有 151 个样例, 即 $U=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_{151}\}$; 5 个条件属性, 即 $A=\{a_1,a_2,\cdots,a_5\}$, 其中, a_1=A native English speaker, a_2=Course instructor, a_3=Course, a_4=A regular semester, a_5=Class size. a_1 表示母语是否是英语, 是一个二值属性; a_2 表示课程讲师, 共 25 个课程讲师, 每个课程讲师用一个符号值表示, 共 25 个值; a_3 表示助教课程, 共 26 门课程, 每门课程用一个符号值表示, 共 26 个值; a_4 表示是否正常学期, 是一个二值属性; a_5 表示班级规模, 是一个数值属性. 显然, TAE 数据集是一个混合类型数据集.

代表性的分类与回归算法包括分类与回归树、神经网络、支持向量机等 [3].

1.1.2 聚类分析

聚类分析处理的对象是没有类别信息的数据集, 即信息表. 聚类分析就是将信息表中的样例划分为若干个簇 (聚类), 使得同一个簇内的样例比不同簇内的样例更相似 [8, 9]. 实际上, 聚类也是一种分类, 只是在聚类的过程中, 没有样例的类别信息可以利用. 由于在聚类过程中没有用到样例的类别信息, 因此聚类分析是一种无导师学习. 聚类算法可分为基于划分的算法、层次算法、基于密度的算法、基于网格的算法和基于模型的算法 [3]. 在聚类分析中, 针对给定的数据集, 选择合适的相似性度量至关重要. 相似性度量可以大致分为基于距离的度量和基于相关性的度量 [10-12]. 设 $\boldsymbol{x}_i, \boldsymbol{y}_j \in \mathbf{R}^d$, 下面介绍几种常用度量 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 相似性的定义, 其他相似性度量的定义可参考文献 [10].

1. 基于距离的相似性度量

(1) 欧氏距离

在基于距离的相似性度量中, 欧氏距离是最常用的, 样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的欧氏距离定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \left[\sum_{k=1}^{d}(x_{ik} - y_{jk})^2\right]^{\frac{1}{2}} \tag{1.1}$$

(2) Manhattan 距离

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Manhattan 距离定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \sum_{k=1}^{d}|x_{ik} - y_{jk}| \tag{1.2}$$

(3) Minkowski 距离

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Minkowski 距离定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \left[\sum_{k=1}^{d}(x_{ik} - y_{jk})^p\right]^{\frac{1}{p}} \tag{1.3}$$

(4) Chebyshev 距离

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Chebyshev 距离定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \max_k |x_{ik} - y_{jk}| \tag{1.4}$$

(5) Camberra 距离

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Camberra 距离定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \frac{\sum_{k=1}^{d} |x_{ik} - y_{jk}|}{\sum_{k=1}^{d} |x_{ik} + y_{jk}|} \tag{1.5}$$

(6) Sorensen 距离

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Sorensen 距离定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \frac{\sum_{k=1}^{d} |x_{ik} - y_{jk}|}{\sum_{k=1}^{d} (x_{ik} + y_{jk})} \tag{1.6}$$

(7) Søergel 距离

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Søergel 距离定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \frac{\sum_{k=1}^{d} |x_{ik} - y_{jk}|}{\sum_{k=1}^{d} \max\{x_{ik}, y_{jk}\}} \tag{1.7}$$

(8) Kulczynski 距离

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Kulczynski 距离定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \frac{\sum_{k=1}^{d} |x_{ik} - y_{jk}|}{\sum_{k=1}^{d} \min\{x_{ik}, y_{jk}\}} \tag{1.8}$$

2. 基于相关性的相似性度量

(1) Pearson 相关系数

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Pearson 相关系数定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \frac{\sum_{k=1}^{d} (x_{ik} - \overline{x}_i)(y_{jk} - \overline{y}_j)}{\left[\sum_{k=1}^{d} (x_{ik} - \overline{x}_i)^2 \sum_{k=1}^{d} (y_{jk} - \overline{y}_j)^2\right]^{\frac{1}{2}}} \tag{1.9}$$

(2) Cosine 相关系数

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Cosine 相关系数定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \frac{\sum_{k=1}^{d}(x_{ik} \times y_{jk})}{\left[\sum_{k=1}^{d}(x_{ik})^2\right]^{\frac{1}{2}} \left[\sum_{k=1}^{d}(y_{jk})^2\right]^{\frac{1}{2}}} \tag{1.10}$$

(3) Jaccard 相关系数

样例 $\boldsymbol{x}_i$ 和 $\boldsymbol{y}_j$ 之间的 Jaccard 相关系数定义为

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \frac{\sum_{k=1}^{d}(x_{ik} \times y_{jk})}{\sum_{k=1}^{d}(x_{ik})^2 + \sum_{k=1}^{d}(y_{jk})^2 - \sum_{k=1}^{d}(x_{ik} \times y_{jk})} \tag{1.11}$$

1.1.3 关联分析

关联分析 [13, 14] 是一种在大规模数据集中寻找关联关系的数据挖掘任务, 关联关系有两种体现形式, 即频繁项集和关联规则. 关联分析最成功的应用是购物篮分析, 在这一应用实例中, 频繁项集是经常被一起购买的物品的集合, 关联规则表示两种物品之间可能存在很强的关系. 下面通过一个例子来解释相关概念.

表 1.5 是某超市的一个交易清单, 集合 {尿布, 啤酒}是一个频繁项集, {尿布, 卫生纸, 啤酒}也是一个频繁项集. {尿布}→ {啤酒}, 是一条关联规则, 它表示如果某人买了尿布, 那么他很可能还买了啤酒.

表 1.5 某超市的一个交易清单

交易编号	购买物品
2015010101	尿布, 啤酒
2015010123	尿布, 卫生纸, 啤酒
2015010137	啤酒, 卫生纸, 纸巾, 尿布
2015010155	卫生纸, 尿布, 啤酒, 牙膏, 毛巾
2015010166	毛巾, 香皂, 洗衣液, 尿布

支持度是定义项集频繁程度的一种度量, 被定义为数据集中包含该项集的记录所占的比例. 在表 1.5 中, 项集 {尿布}的支持度为 $\frac{5}{5}$, 项集 {尿布, 啤酒}的支持度

为 $\frac{4}{5}$, 项集 {尿布, 卫生纸, 啤酒}的支持度为 $\frac{3}{5}$. 可以定义一个支持度阈值, 称支持度大于这个阈值的项集为频繁项集.

置信度是关联规则可信程度的一种度量, 关联规则 {尿布}→ {啤酒}的置信度定义为项集 {尿布, 啤酒}的支持度与项集 {尿布}的支持度的比值, 其置信度为 $\frac{4}{5}$. 显然, 关联规则 {尿布, 啤酒}→ {卫生纸}的置信度为 $\frac{3}{4}$.

说明: 支持度也可以从关联规则的角度来定义, 如关联规则 {尿布}→ {啤酒}的支持度定义为项集 {尿布, 啤酒}的支持度.

下面给出关联分析问题的形式化定义. 设 $I=\{I_1,I_2,\cdots,I_m\}$ 是包含 m 个项目的集合, $T=\{T_1,T_2,\cdots,T_n\}$ 是交易记录的集合, 也称为事务数据库. 其中, T_i 是 I 的非空子集. 给定 $X,Y\subseteq I$, 而且 $X\bigcap Y=\varnothing$, 关联规则是形如 $X\to Y$ 的蕴含式. 关联规则 $X\to Y$ 的支持度是指数据库 T 中包含 $X\bigcup Y$ 的百分比. 置信度是指包含 X 的记录中同时也包含 Y 的百分比, 也就是条件概率 $P(Y|X)$. 如果规则 $X\to Y$ 的置信度和支持度都高于某个阈值, 则认为项集 X 和 Y 具有关联性.

代表性的频繁项集挖掘算法包括 Apriori 算法、FP-Tree 算法、HotSpot 算法等 [3].

1.1.4　时间序列分析

时间序列分析是用已有的数据预测未来发展趋势的一种数据挖掘方法. 在时间序列分析中, 数据的属性值是随时间变化的 [3]. 回归分析不强调数据间的先后顺序, 而时间序列分析要考虑时间特性, 尤其要考虑时间周期的层次, 如天、周、月、年等. 时间序列分析在经济发展预测、股票趋势预测、公司销售额预测等方面有重要应用.

影响时间序列的变化因素包括以下四种.

① 长期趋势因素. 反映的是一个较长时间内的发展方向, 如在经济发展中, 可以在一个相当长的时间内表现为近似直线的发展趋势.

② 季节变动因素. 反映的是受季节变化的影响, 形成的一种长期或幅度固定的周期性波动.

③ 周期变动因素. 反映的是受各种因素的影响, 形成的上下起伏不定的波动.

④ 不规则变动因素. 反映的是受各种偶然因素影响, 形成的不规则变动.

常用的时间序列分析方法包括回归预测方法、趋势外推法、指数平滑法等.

1.1.5　偏差检测

偏差检测是用来发现与正常情况不同的异常和变化, 并分析这种变化是有意的欺诈行为, 还是正常的变化 [3]. 在有些数据挖掘应用中, 将这些异常信息作为噪声