摘　要

随着科技的不断发展，在科学研究和日常工作中人们经常会遇到大量的高维数据，它们蕴含了极为丰富和相近的客观信息。如何直观的表示这些海量数据，或者从中获取用户感兴趣的隐藏的信息与规律，一直是学者们不断追求的目标。可视化技术能将数据信息转变为直观的、以图形或图像形式表示的、随时间空间变化的物理现象或物理量呈现在用户面前，使用户能观测到传统意义上不可见的事物或现象，可视化方面的研究迅速发展。

本文首先对信息可视化技术和多维可视化的常用方法进行研究，对各种技术的原理、特点以及交互手段进行了描述。其中在可视化过程中对于在数据可视化维度控制显示方面往往缺乏良好的指导，需要用户根据经验逐步试探性的对维度的排列顺序以及用于可视化显示的维数进行控制，这样一些重要规律可能被忽略。本文利用维相似度算法对数据维的排列顺序进行规划；采用属性相关分析算法对参与显示的维数进行控制。通过实验验证上述算法提高了可视化的显示效果。

文章最后给出了一个篮球运动员技术指标分析系统的设计和重点部分的实现。篮球运动员的技术指标分析的内容具有多维性，并且多维数据存在一定的关系，用户需要动态地改变对比分析的内容，要求能够从多角度地展示数据，快速、准确地得到结果。本系统实现了多种可视化方法，可以使用户从不同侧面分析理解数据，通过简单的交互过程得到所需的可视化结果，并进一步验证维相似度算法和属性相关分析算法对多维数据可视化显示效果的提升，使用户更为便捷的对数据进行观察与分析，获得有效的信息。

关键词：多维数据可视化，维数控制，维相似度，信息增益，图表

The Research and Application of Multidimensional Data Visualization in Data Mining

Abstract

With the development of database technology and the popularization of database application, the quantity of data that is stored in computer is being huger and huger increasingly. People want to analyze the data so they can obtain knowledge or information, instead of just managing them. Information visualization technology is one of important implements to display data, which can discover the relation between information and latent characteristic. Multi-dimension data visualization is a focus content of information visualization field.

First, this paper make a summary of information visualization technology and some conventional methods for visualizing multi-dimension data, having an introduction about the principle and characteristic of various technique. In process of visualization the arrangement of dimensions are lacks of guiding; so many knowledge and information will be overlooked. In this paper a arithmetic of dimension similarity is applied. In the first step, the similarities of all the dimensions are calculated, and then a similarity matrix is built with these values of dimension similarity. At last an arrangement of dimension is gained with the matrix and optimization arithmetic. When the quantity of dimension is too large, it is hard for user to watch and understand the data. So the information gain arithmetic is used. Entropy is applied to account the information gain, and then series dimensions that have a low information gain are deleted from the visualization. Thus the users can find the knowledge and rules more easily. Both these researches get good experiment results, so they are applicable.

In the last part of the article there is a practice on basketball player data investigation statistics analysis system. The contents of analysis are multi-dimensional and user want to change the contrast items dynamically, demanding display the data from different aspects and getting accurate result rapidly. This system combined conventional data visualization method with parallel coordinate technology to deal with data availably. User can get the visual result by operating interactively and analyze the results of many kinds of situations under parallel coordinates, reducing their workload greatly.

Key Words: Multi-dimension data visualization, Controling of dimension quantity, Dimension simlarity, Information gain, Chart

独 创 性 说 明

本人郑重声明：所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写的研究成果，也不包含为获得沈阳工业大学或其他教育机构的学位或证书所使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中做了明确的说明并表示了谢意。

签名：张文鹤 日期：2007.1.2

关于论文使用授权的说明

本人完全了解沈阳工业大学有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。

（保密的论文在解密后应遵循此规定）

签名：张文鹤 导师签名：[illegible] 日期：2007.1.2

1 引 言

1.1 研究背景及意义

计算机应用于数据处理已经有 40 多年的历史，由于受到计算机发展水平的制约，数据只能以批量处理却不能进行复杂的交互，更不能对信息进行干预及导引。人们已经不能满足于等待计算机结果的输出。用户希望能依靠计算机强大的计算能力获取蕴含在海量数据中的信息与模式。这种方式不仅不能得到有关数据的直观、形象的整体概念，还可能丢失大量信息。

由于缺乏有效分析手段，常常不得不割舍庞大数据群中的大部分有用数据，导致应用的信息处理精度降低。海量数据的产生与不能有效解释这些数据的矛盾日益尖锐。因而，迫切要求提供一种能够处理和解释这些海量数据的技术，科学计算可视化就是顺应这一要求而产生的。

随着社会信息化的推进和网络应用的日益广泛，信息源越来越庞大。除了需求对海量数据进行存储、传输、检索以及分类等外，更迫切需求了解数据之间的相互关系及发展趋势。实际上，在激增的数据背后，隐藏着许多重要的信息和模式，人们希望能够对其进行更高层次的分析，以便更好地利用这些数据。这一需求促成了数据挖掘技术的发展。它是从大量的、不完全的、有噪声的、模糊的、随机的数据中，提取隐含在其中的、人们事先不知道的、又是潜在有用的信息和认识的过程[1,2]。

为了使数据挖掘系统发现知识的过程和结果展示易于理解和在发现知识过程中进行人机交互，要发展发现知识的可视化方法。为了了解数据之间的相互关系及发展趋势，人们可以求助于可视化技术。在这种背景下，数据可视化技术获得人们越来越高的重视和高速的发展。它凭借着计算机强大的计算能力和图形图像处理能力，将大型的数据记录集转化为静态或动态的图形或图像呈现在用户面前，并允许通过交互手段控制数据的抽取和画面的显示，使隐含在数据中的不可见的规律和模式在用户面前加以呈现，为人类分析数据、理解数据和寻找规律做出决策提供了有力的手段。

数据可视化涉及到计算机图形学、图像处理、计算机视觉、计算机辅助设计等多个领域[2,3]，成为研究数据表示、数据处理、决策分析等一系列问题的综合技术。近一些年以来，

可视化技术受到越来越多的重视，成为数据分析较为有效的方法之一。通过本课题的研究，深入了解目前较为普及的数据可视化技术的原理以及相关的技术，掌握它们在可视化过程中的一些尚待解决的问题以及一些较为成熟的实施方法。在实际应用中利用本课题的成果，更加有效的对多维数据进行合理分析和处理，生成符合人类认知系统的直观图形，从而使人从图形中高效的获取有用的知识和信息。因此在信息时代的大潮中，对数据可视化技术进行深入研究并进行相应的改进与实践，可以大大简化相关系统开发过程，提高数据分析处理的效能，提升决策支持的准确性。

1.2 研究对象及内容

科技的迅速发展，人们对于基础科学的研究越来越深入，如医学、气象学和流体力学等[5,6]。信息获取技术和数据存储技术的不断提高，也使人们研究和分析这种大型复杂数据成为可能。例如：在医学或生物学中经常需要分析被观测者的身高、体重、身体内各微量元素的含量以及身体各脏器的状况等变量。这些由多个变量描述现象的数据，抽象出来就是一种高维数据。多维数据的广泛使用为用户掌握丰富的客观现象、获取详细的信息提供了有利的途径，但是随着数据维度的大幅度提高也给后续的数据处理工作带来了巨大的困难。本课题的研究对象就是高维数据，旨在对目前已有的可视化技术进行深入研究，找出各种方法存在的一些不足与困难并加以改进，从而使可视化技术更加充分的应用于多维数据显示领域，为用户对多维数据进行处理和分析通过更有力的帮助。

我们的研究内容主要包括：在数据可视化过程中采用与聚类算法相似的方法计算各个维度之间的相似关系，并通过优化算法找到信息增益最大的规划方法对维度的排列进行合理的排列；在图标以及几何变换过程中，对图形图线的颜色采用与主题相关的映射方式；在信息技术迅速发展的今天如何利用已有的技术便捷的实现支持多维数据可视化系统一些方法和技术。

1.3 国内外研究现状

国外从20世纪80年代末提出可视化技术以来，对它的研究已经取得了相当大的进展。可视化的应用范围不断扩大，除了众多的科学和工程领域，在商业和日常生活中也得到越来越多的应用。研究者已经建立了可视化实验室、可视化专题讨论、可视化国际

会议以及可视化教育来促进可视化的教育和发展。许多大学、研究机构和国家实验室对可视化工具、环境和应用等方面展开了广泛而深入的研究。目前可视化技术的发展还结合了超级计算机、高速网络、高性能图形工作站和虚拟现实技术，同时在市场上也推出了许多可视化软件产品。

我国科学计算可视化技术的研究始于 90 年代初。由于数据可视化所处理的数据量十分庞大，生成图像的算法又比较复杂，过去常常需要使用巨型计算机和高档图形工作站等。因此，数据可视化开始都在国家级研究中心、高水平的大学、大公司的研究开发中心进行研究和应用。近年来，随着 PC 功能的提高、各种图形显卡以及可视化软件的发展，可视化技术已扩展到科学研究、工程、军事、医学、经济等各个领域[6,7]。比如，我国“863”高技术发展研究课题——数字化虚拟中国人数据集构建与海量数据库系统，它运用人体信息和计算机技术，将真实的人体断层数据进行处理，为不同行业提供后续开发虚拟人体的数据参数。它使用计算机在三维空间来模拟真实人体的所有特征，这就是可视化技术的一个典型应用。随着 Internet 的兴起，信息可视化技术更是方兴未艾。虽然国内部分大学和科研机构正在研究可视化算法、移植或开发各种可视化工具，且在油气勘探、气象、医学、流体力学等领域的应用方面取得了一大批可喜的成果。但从总体上来说，国内不仅在硬件上，同时在应用方面与国外先进水平差距较大[8,9]，特别是在商业软件方面还基本处于空白[10-12]。因此，组织力量开发可视化商业软件，并通过市场竞争，促使其逐步成熟，已成为当务之急。这也给我们对该技术的研究提供了广阔的发展空间。

1.4 论文组织结构

本文的结构安排如下：

第 1 章为引言部分，主要介绍研究的背景及意义，分析了国内外关于多维数据可视化的发展动态，确定了研究的对象和内容。

第 2 章详细阐述了可视化的一些基本概念，并对可视化模型的框架进行研究，提出了在可视化系统设计中需要考虑一些心理学方法着重利用计算机来模仿人的认知系统去设计数据的显示方式，可以避免信息的歪曲，适应人的感知系统。最后介绍数据挖掘

系统中所研究的多维数据模型，澄清了一些数据挖掘系统与普通信息管理系统之间的概念差别，为在数据挖掘系统中应用多维数据可视化技术打下良好的基础。

第 3 章针对多维数据可视化技术进行深入研究，首先逐一介绍目前较为成熟的多维数据可视化技术，并从多个角度对这些算法进行比较，分别指出在针对不同数据时，这些算法的优缺点；然后介绍了可视化过程中的一些交互技术，并指出多维数据可视化过程中维数控制问题提出一些观点。

第 4 章更加深入的研究多维数据可视化时维数控制算法，提出以维相似度算法来指导多维数据在可视化时，维度排列的问题；提出结合数据挖掘过程中，针对概念描述和分类问题的 AOI 算法对多维数据进行维度数量及内容的控制。

第 5 章通过介绍了一个篮球运动员技术指标分析系统实现方法，对可视化方法中的几何变化法和图标法进行实践，验证了多维数据可视化中维度控制的可行性。

第 6 章是全文的结论部分，对全文进行总结。

2 多维数据可视化的基本概念

2.1 可视化的概念

可视化是一系列的转换，这种转换将原始模拟数据转换成可显示的图像，这种转换的目的在于将信息转换成可被人类感应系统领悟的格式，用于利用计算机图形来加强信息的传递和理解。

可视化的基础是计算机图形学，目前它已经发展成为研究用户界面、数据表示、处理算法和显示方式等一系列问题的一个综合性学科。根据侧重面的不同，可视化可以分成三个分支：科学计算可视化、数据可视化和信息可视化。

科学计算可视化是把计算中涉及的和空间变化的物理现象或物理量呈现在研究者面前，是他们能够观察到模拟和计算的过程，使其看到传统意义上不可见的事物或现象；同时还提供与模拟和计算的视觉交互手段；数据可视化比科学计算可视化具有更加广泛的内容，它不仅包含工程领域数据的可视化，还包含其他领域（如经济、金融、商业等）中数据的可视化。在科学研究过程中，科学家们不仅需要通过图形图像来分析由计算机产生和获取的数据，而且还需要了解计算过程中数据的变化。科学计算可视化可以实现对计算和编程过程的引导和控制，通过交互手段改变过程所依据的条件，并观察其影响。数据可视化技术指的是运用计算机图形学和图像处理技术，将数据转换为图形和图像在屏幕上显示出来，并进行交互处理的理论、方法和技术。它涉及到计算机图形学、图像处理、计算机辅助设计、多媒体技术、虚拟现实技术、计算机视觉以及人机交互等多个领域的知识；随着互联网络技术和电子商务的发展，数据的规模一再增大，为了获取在数据中隐含的大量信息与知识，人们对信息可视化的需求愈发强烈。信息可视化的本源仍然是数据可视化，人们可以通过数据可视化技术来发现大量金融、通信和商业信息数据中的隐含规律，从而为决策提供依据与支持。在科学计算可视化中，显示的对象涉及标量、矢量等不同类别的空间数据研究重点放在真实快速显示三维数据场，而在信息可视化中，显示的对象主要是多维的标量数据。

科学技术在不断发展，计算机在各个领域的应用也在不断深入。无论是科学计算、企业生产运作、公司的商业活动都是以海量的数据操作为基础的。现有的大部分信息管

理系统虽然可以为用户提供基于海量数据的各式各样的查询报表，但这仅仅停留在对数据的重组层面，并没有从根本解决数据的表现，很难进行决策分析。实际上用户需要的不仅是拥有数据，更重要的是“看到”数据，即数据的“可视”[13,14]。为了实现数据的可视以支持决策分析，众多科学家做出了努力也取得了很多成果。

2.2 数据可视化的框架与模型

数据可视化技术宗旨是帮助用户精确的发现蕴含在海量数据中的信息，并降低一些客观因素对于数据的影响。理想的可视化方法可以帮助用户在观察数据的同时获得具有洞察力的推论。由于数据可视化具有这种潜力，因此在数据挖掘与探索、信息重获、策略分析以及战略智能领域都得到广泛的应用[15]。

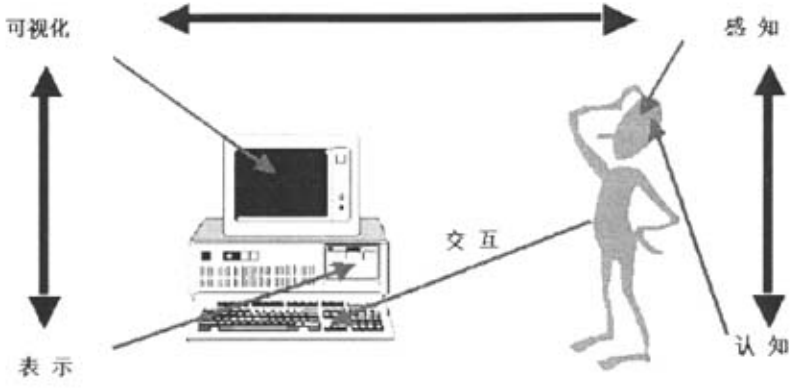


图 2.1 数据可视化心理学框架，基于 Lee 和 Vickers 理论

Fig. 2.1 A psychological framework for data visualization. Base on Lee and Vickers

感知是人类认识和了解世界的主要手段，图像是由理解产生的脑海中的画面。感知是由各个部分彼此间的关系的建立起来的一个有意义的整体。人类在事物中找出模式的能力和把各个部分整合为一个有意义的整体的能力是人类思考和感知的重要手段。当人观察环境时，实际上是在进行一项非常复杂的任务：从独立的、不同的感官元素得出本质上的意义。人类的眼睛不像照相机，它不是一个专门捕捉图像的机器，而是一个能检

测到变化、模式、特征的复杂的处理单元。当人观察周围的三维环境时，一些属性如轮廓、质地和一些规律特征能让人区别对待物体。人类一般情况下不会根据这些属性的值进行推理，他们往往需要利用图形图像来完成从感知系统获取数据特征到认知系统获取有趣信息的过程，这种框架就是一种心理学框架，如图2.1所示。

在数据可视化过程中，考虑一些心理学方法着重利用计算机来模仿人的认知系统去设计数据的展示已被广泛认可。为了产生精确高效的理解，避免信息的歪曲，可视化技术必须适合人的感知系统。然而问题在于开发数据可视化系统时，更为抽象的感知过程与认知过程没有直接的联系，为了使数据的分析与操作更为有效，系统所转达的信息结构必须要兼容数据表示的需求以及人类人之过程的偏好。

在图2.1中显示了一个符合心理学框架的可视化系统需要包括：感知组件和认知部件。这种框架产生的目的是为了使用户以一种类似频道的方式来观测人工系统中的数据信息，并充分调动用户的主观认知过程。人工系统主要反映客观事实，对于自然现象进行纯客观的展示，从这种意义上说它与人的感知系统具有良好的兼容性，两者都是通过对客观事实出发对事实进行处理和计算，最终得出结果；而人的认知系统与上述过程不尽相同，它是根据人的感知系统获取的信息在大脑中形成的影响或认识为基础，对事实进行重现并根据脑海中的印象对抽象的数据进行理解分析，抽取出有用信息与知识的过程。由于人工系统与人的认知过程存在差异，因此需要充分区别两者不同，在系统中合理安排对数据的显示方式，使之充分适应人的认知过程，从而在知识发现与决策支持过程中获取更高的效能[15]。

数字信息时代,网络和各种现代化的电子通信设备的飞速发展造成数据流呈指数倍数增长。这些激增的数据背后隐藏了大量潜在有用的知识。数据的走向有两种:一种是由数据最终变成数据垃圾，另一种则是由数据变成信息，最终变成知识指导人们做出决策。决定数据最终出口的关键在于有效的信息抽取方法和知识发现手段。

然而，信息大潮的冲击，使人们在很多应用中需要用到很大规模的数据库系统，这些数据库的数据量动辄几百万条，维度达到几十甚至几百。在面对这些庞大而且复杂的数据时，领域专家一直致力于解决诸如：应该从哪里入手，什么看上去是有趣的，是否

还有其他可用的数据等问题。事实证明，在这些大型数据集和数据库基础上获取有用信息的过程中，采用可视化计算和操作是比较理想的选择。一些数据挖掘技术和算法在使用中难于被决策者理解和使用，而可视化可以使数据和挖掘结果更容易理解并允许对结果进行比较和检验，因此在知识发现、决策支持系统中采用符合人类认知过程可视化技术可以加强数据挖掘处理的效能，对数据挖掘系统是非常有帮助的；另外可视化模型还可以兼容数据挖掘算法，并指导数据挖掘过程。

2.3 多维数据模型

目前较为流行的数据仓库与 OLAP 工具大多基于多维数据模型。该模型将数据看作数据立方体(Data Cube)形式，如图 2.2 所示。采用此种方式组织数据可以使数据仓库系统高效管理大量历史数据，提供汇总和聚集机制，并在不同粒度级别上存储和管理信息，便于系统利用数据做出合理的决策。

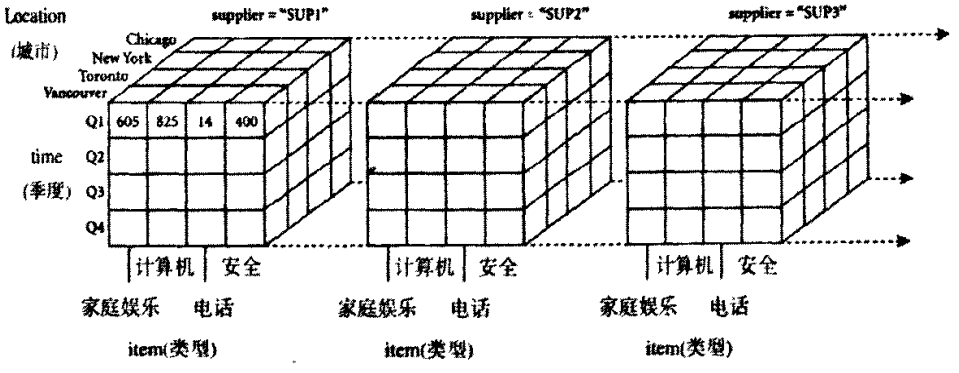


图 2.2 数据立方体实例，图中是一个销售数据的 4-D 数据立方体表示，维是 time，item，location 和 supplier，所显示的度量为 dollars_sold（单位：$1000）

Fig. 2.2 An instance of data cube, there is a 4-D data cube whose dimensions are time, item, location and supplier. The measurement is $1000

数据立方体模型允许以多维对数据建模和观察，它是由维和事实所定义。一般地，维是关于一个组织想要记录的透视或实体。每一个维都有一个表或者表中字段与之相关

联，可以利用它对于数据维进行进一步描述。该维表或者维字段可由用户或专家设定，或者根据数据分布自动产生和调整。

通常，多维数据模型围绕中心主题组织。该主题用事实表示。事实一般采用数值度量。把它们看作数量，是因为用户利用它们分析维之间的关系。事实表包括事实的名称或度量，以及每个相关维表的关键字。

实体-关系数据模型广泛用于关系数据库的设计。在那里，数据库模式由实体的集合和它们之间的联系组成。这种数据模型适用于联机事务处理(OLTP)。然而，数据仓库需要简明的、面向主题的模式，便于联机数据分析[16]。目前较为流行的数据仓库模型是多维数据模型。这种模型可以以星形模式、雪花模式和事实星座模式存在。

星形模式(Star schema)：最常见的模型范例是星形模式，其中数据仓库包括：(1)一个大的包含大批数据和不含冗余的中心事实表，(2)一组小的附属维表，每维一个。这种模式很像星星爆发，维表围绕中心表显示在射线上，如图 2.3 所示。

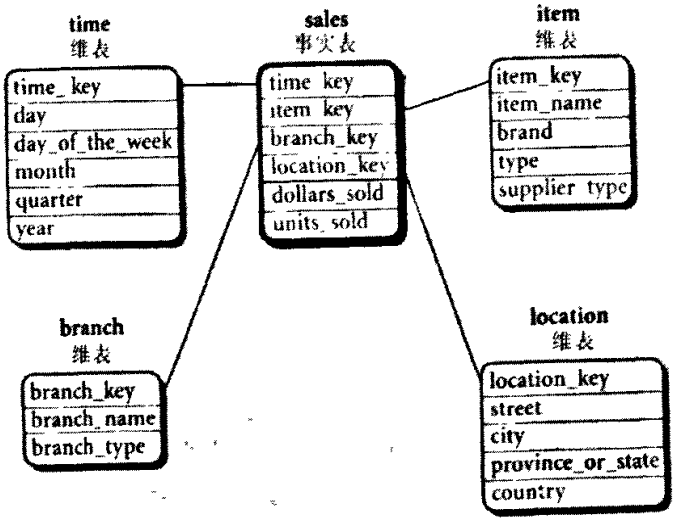


图 2.3 星型模式数据模型

Fig. 2.3 Star schema

雪花模式(snowflake schema)：雪花模式是星形模式的变种，其中某些维表是规范化的，因而把数据进一步分解到附加的表中。模式图形成类似于雪花的形状如图 2.4 所示。

事实星座(fact constellation)：复杂的应用可能需要多个事实表共享维表。这种模式可以看作星形模式集，因此成为星系模型(galaxy schema)，或事实星座，如图 2.5 所示。

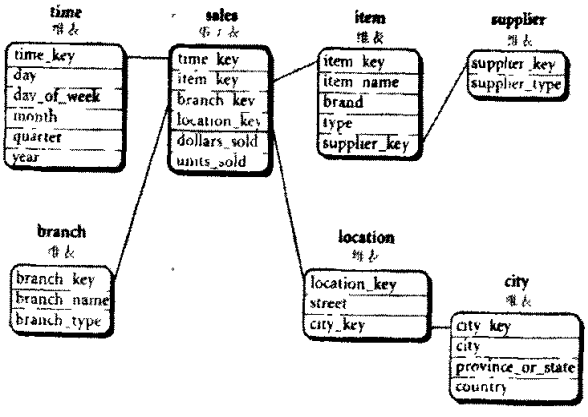


图 2-4 雪花模式数据模型

Fig. 2.4 Snowflake schema

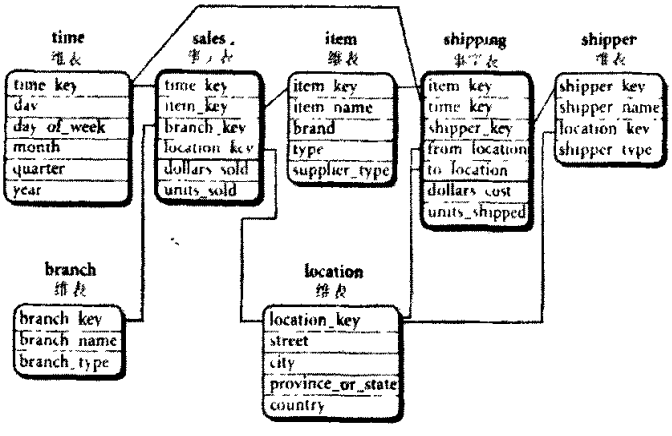


图 2-5 事实星座数据模型

Fig. 2.5 Fact Constellation

采用上述的数据模型，使数据组织成多维的形式，每个维度还可以根据所表示的事实抽象为多个层次，这种组织数据的方式可以使用户从不同角度灵活的观察数据，用户可以从数据立方体的各个方向获取视图，并进行交互查询和相关操作。对于数据模型的常用操作包括：上卷(roll-up)、下钻(drill-down)、切片(slice)、切块(dice)以及转轴(pivot)等，可以参见文献[1]。上卷操作是通过一个维的概念分层向上攀升，以得到更加笼统或综合的事实；下钻操作是上卷的逆操作，它由不太详细的数据转换到更加详细的数据信息，它沿维的概念分层向下或引入新的维来实现；切片操作在给定的数据立方体的一个维上进行选择，产生一个二维的平面；切块操作通过对两个或多个维执行选择定义子立方体；转轴是一种目视操作，它转动数据的视角，提供数据的替代表示。以这种方式组织数据不仅对数据挖掘中概念描述和比较有很大好处，同时对数据的可视化操作同样大有裨益的。在可视化过程中，采用这种多维数据模型，使数据都能够按照主题进行分类分层，不仅可以降低显示与挖掘的复杂性，而且更加符合人认知系统的特点，更便于用户对数据挖掘及可视化结果的理解。

需要指出的是数据挖掘应当是以人为中心的过程。用户将经常与系统交互，进行探测式数据挖掘，而并不特别要求数据挖掘系统自动产生模式与知识，因此采用数据立方体数据模型指导数据挖掘过程是具有很强实用性的。

3 数据可视化技术常用方法

3.1 数据可视化的分类

数据的可视化会涉及到数据类型、可视化技术及数据进行交互和变形的技术。所有的三个要素构成了对数据的可视化。图 3.1 描述了三个要素各自所包含的内容。

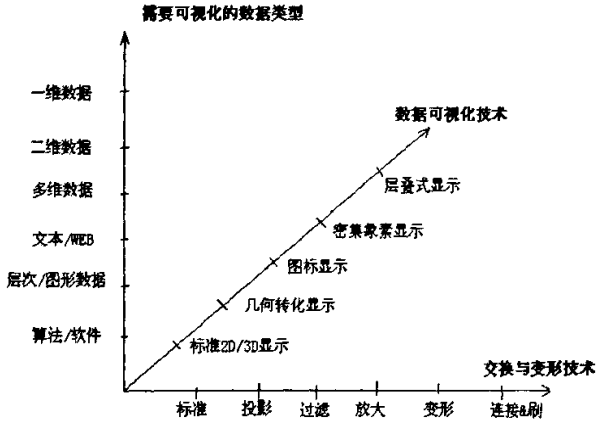


图 3.1 数据可视化的三要素

Fig. 3.1 Three factors of visualization

可视化数据类型(Data to be Visualized)

(1)一维数据只有一个维度。典型的一维数据的实例是时序数据。在每一个时间点有一个或多个数据值相关联可以参见文献[12]。

(2)二维数据有两个不同维。典型的实例是地理数据，有经度和维度两个不同的维。可以采用二维坐标系进行显示。尽管表面上处理时序或地理数据等一维/二维数据，但是当数据良很大时，这种方法不是和容易理解数据。

(3)多维数据集包括超过三个的属性，这样不能简单的作为二维或三维数据来显示。多维数据模型的实例是关系数据库中的表，表的每一列都表示一个属性。采用平行坐标、象素显示、散点图矩阵技术等方法对数据集进行显示和描述，参见文献[16]。

(4)文本和超文本数据是网络时代的一种重要的数据类型[17],这些数据不能轻易的被描述为数字，因此许多标准的可视化技术不能被应用。一般情况下，首先要把数据转化为向量描述，然后再应用可视化技术。

另外，还有一些数据类型，如图形、层次数据、算法和软件等都有专门的一些可视化方法，如：图形可以表示一般数据之间的内部依赖关系；算法和软件的可视化目的是为了帮助对算法的理解，以此来支持软件的开发，如流程图、代码结构图等。

3.2 多维数据可视化技术

如图 3.1 所示，数据可视化技术(Data Visualization Technique)包括几个方面，以下将会逐个介绍每种可视化技术。

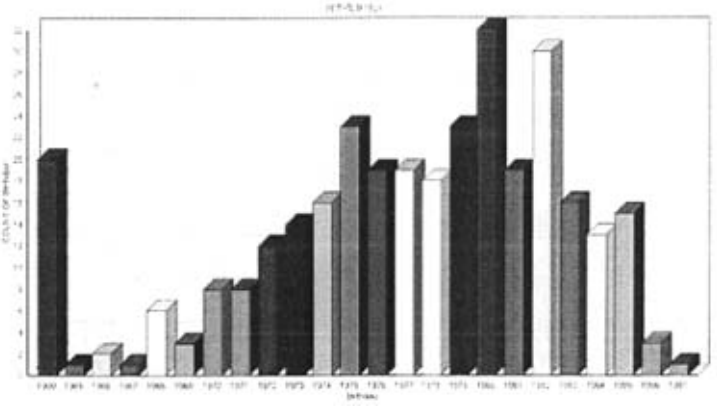


图 3.2 采用条状图对统计数据进行可视化显示

Fig. 3.2 The visualization of Bar Charts

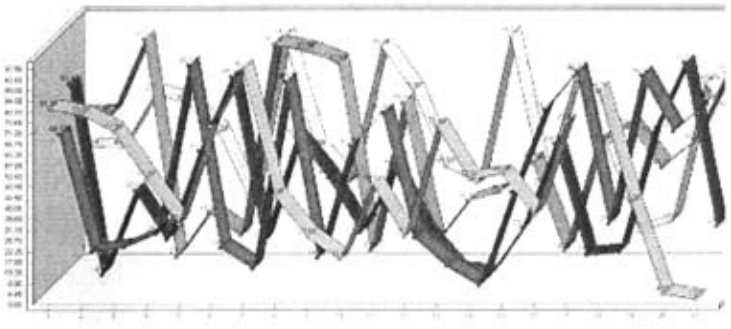


图 3.3 采用线条图对数据进行可视化显示

Fig. 3.3 The visualization of Line Charts

标准的 2D/3D 可视化技术，如二维坐标/三维坐标，条形图(Bar Charts)，线条图(Line Graphs)等等，这些也是用户较为经常使用的数据可视化表达方式，如图 3.2，3.3 所示。

3.2.1 几何转化显示技术(Geometrically-Transformed Displays)

几何转化显示技术旨在发现多维数据集的有趣的转化。目前主流的几何显示技术研究主要包括三种：

（1）散点图矩阵(Scatterplots matrices)：散点图可能是最流行的数据挖掘可视化工具，它可以帮助用户发现簇及其外层，趋势和关系[31]。掠过的点和分类着色的点被用来获得对数据的额外的洞察。当数据点过多，彼此交迭或数据的分解使大量的数据点位于同一个坐标系，放大，扫视全景，抖动就可被用来提高视图效果。当要显示的维数较多时，散点图就很难表现出好的效果了。散列图矩阵解决了这个问题。它使散点图用矩阵的方式排列以表达多维数据集属性彼此间的关系。图 3.4 显示了一个数据集的散列图矩阵。

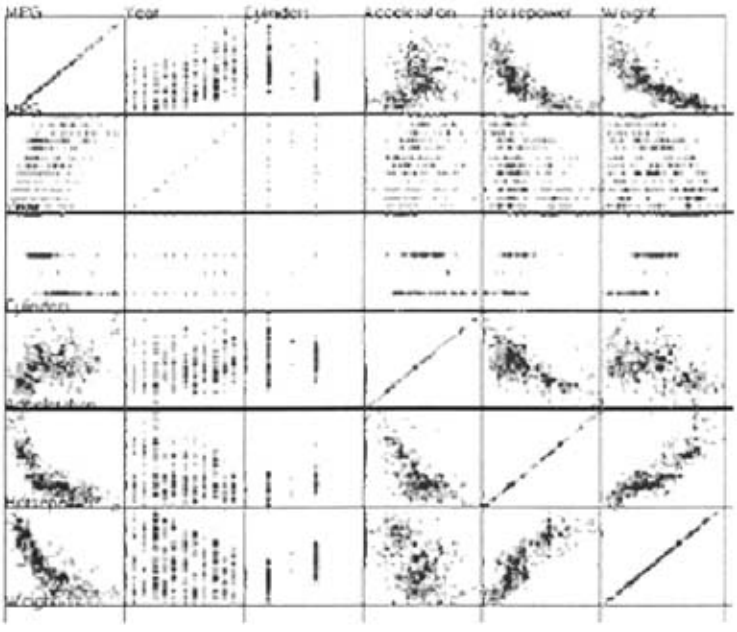


图 3.4 数据集的散列图矩阵

Fig. 3.4 Data set of Scatterplots matrices

（2）解剖视图（Prosection Views）：把截面（section）和投影（projections）组合起来称为解剖（Prosections），这样就可以显示中间维的结构面貌[18,19]。投影能够容易的显示低维的结构。截面能够容易的显示较低的余维数，例如具有高维对象的子空间的交集。图 3.5 显示了一个解剖视图的例子。

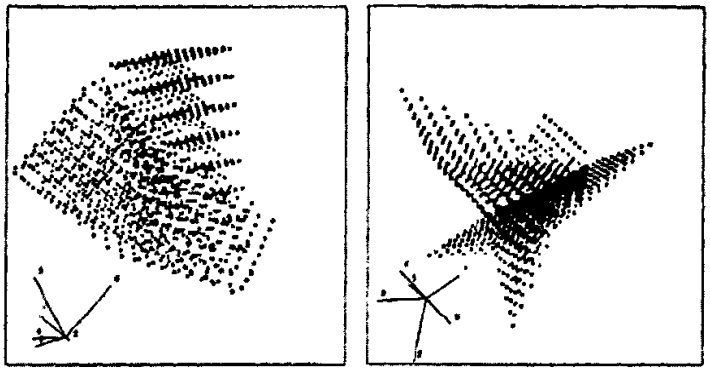


图 3.5 对超度量的空间轨迹二维投影的解剖视图。在右下角上显示了坐标系单元向量的投影

Fig. 3.5 Two 2-Projections of the Ultrametric Locus.(The projections of the coordinate unit vectors are shown in the bottom left corner)

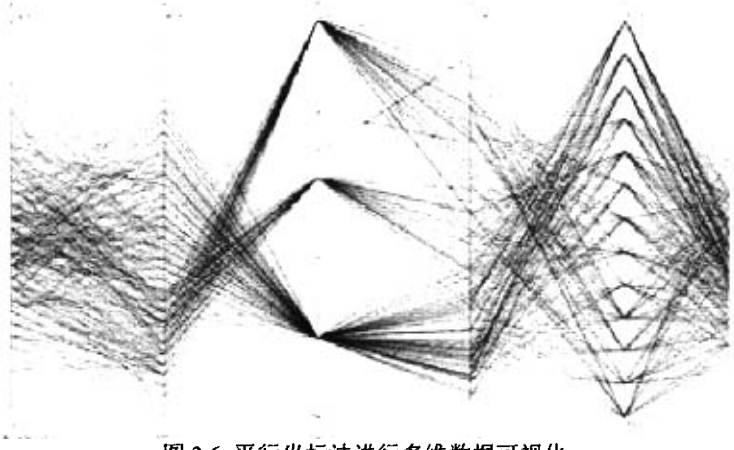


图 3.6 平行坐标法进行多维数据可视化

Fig. 3.6 The Visualization of Parallel Coordinate Technology

（3）平行坐标法（Parallel Coordinates）：平行坐标法是最早提出的在二维平面上显示 n 维空间的数据可视化技术之一，参见图 3.6。它的基本思想是将 n 维数据属性空间用 n 条等距离的平行轴映射到二维平面上，每条轴线对应一个属性维，坐标轴的取值范围从对应属性的最小值到最大值均匀分布[14,20-22]。这样，每一个数据项都可以用一条折线表示在 n 条平行轴上。这个视图能够使用户对每个属性的数据分布有一个粗

略的认识，尤其，不同类型数据以不同颜色显示能够更清晰的表示不同类型数据之间的差异[26]。

3.2.2 图标显示技术(Iconic Displays)

图标显示技术是基于图标的技术，其核心思想是把每个多维数据项画做一个图标。图标可以被任意定义，它们可以是“Chernoff脸谱图”、“针图标”、“星图标”、“棍图标”，这些都是曾经被人们用过的图标形状，参见文献[15,23-25]。例如，在星图标显示技术中，每一维数据用一条射线表示，数据的大小由射线的长短来表示，属性的个数就是射线的条数，所有射线起点相同，彼此夹角也相同，射线的端点由折线段彼此相连。图3.7分别显示了一个星图标和一个脸谱图标的例子，图中显示的是20个具有14维度的数据。实践表明采用图标技术对数据进行可视化，可以充分的将数据各个维度的信息加以显示，使用户可以非常便捷的比较数据间的差异，发现有趣的数据关系。不仅如此，曾有专家对可视化效果进行实验，相同的用户群体对图标可视化结果与其他可视化结果进行观察，结果用户投放在图标可视化结果上的时间超过其他可视化结果的50 %以上。由此可见，采用图标技术进行多维数据可视化可以大幅提升用户的关注度，进而提升系统的可用性。

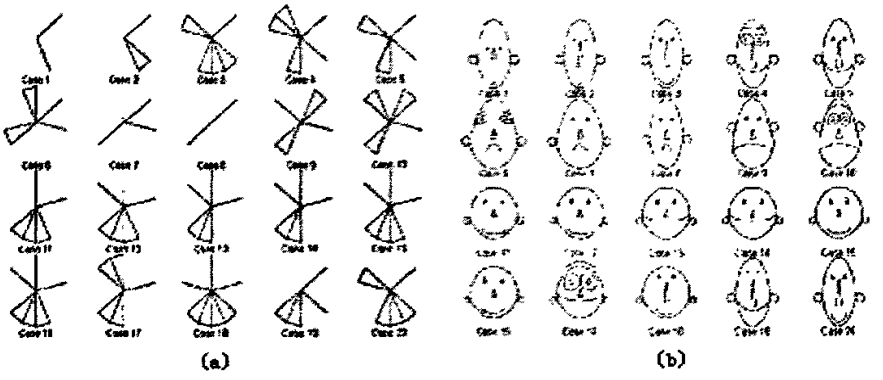
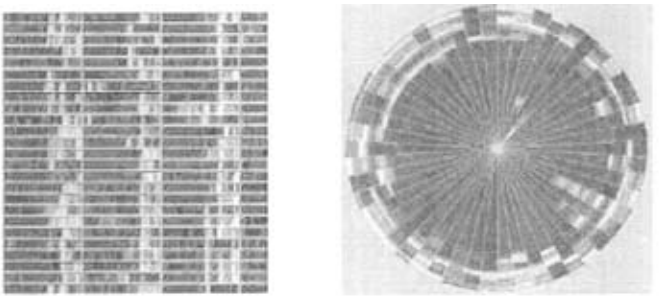


图3.7 对于同一组数据集进行图标可视化。(a)星形图标可视化。(b)Chernoff脸谱图标可视化

Fig. 3.7 Iconic Displays with the same set of data set, (a) is star visualization and (b) is Chernoff face

3.2.3 密集象素显示技术（Dense Pixel Displays）

密集象素技术的基本思想是把每一维数据值映射到一个彩色的象素上，并把属于每一维的象素归纳入临近的区域。因为密集象素显示技术用每一个象素相应的显示每一个数据值，所以此技术允许可视化大量的数据，目前大概能够在同一屏幕上显示超过 1，000，000 个数据值。如果每个数据值由一个象素表示，那么主要的问题就是如何在屏幕上安排这些象素。密集象素技术针对不同目的采取不同的方式安排象素，显示的结果可以对局部关系，依赖性和热点提供详细的信息。著名的例子是递归模式技术(Recursive Pattern Technique)和圆周分段技术(Circle Segments Technique)。递归模式技术基于普通的递归来回地安排象素，其目标尤其在于按照一个属性以自然的顺序表示数据集，用户可以为每个递归层指定参数，随之可以控制象素的安排，以形成语义上有意义的子结构。圆周分段技术的思想是将圆周分成若干部分，每部分对应一个属性。在每部分中，每个属性值由一个有颜色的象素显示，参见文献[27,28]。图 3.8 分别显示了递归模式技术与圆周分段技术的数据可视化结果。



（a）　　（b）

图 3.8 密集象素数据可视化技术。（a）递归模式技术可视化，（b）圆周分段技术可视化。

Fig. 3.8 The visualization of the Pixel-oriented technique. (a) is Recursive Pattern Technique and (b) is Circle Segments Technique

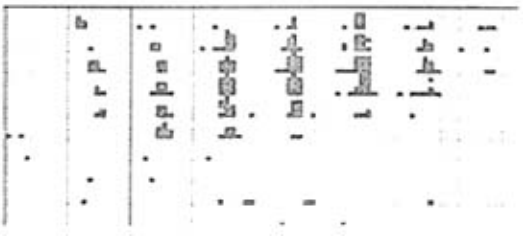


图 3.9 基于层次式显示技术的可视化结果

Fig. 3.9 Stacked Displays

3.2.4 层叠式显示技术(Stacked Displays)

层叠式显示技术以分层的方式将数据分开表示在子空间中。将 N 维属性空间划分成二维平面上的子区域，子区域彼此嵌套，基本思想是将一个坐标系统嵌入到另外的坐标系统中，属性数值被划分成几个类。视图的产生是通过将最外层坐标系统分成矩形单元，在这些单元中，接下来的两个属性通常会横跨第二层坐标系统。结果视图的有效性很大程度上依赖于外层坐标上数据的分布。因此，用来定义外层坐标系统的维数必须仔细的选择。一个首要的规则是首先选择最重要的维，参见文献[29]。图 3.9 是一个石油挖掘数据的维数层叠视图，其中，纬度和经度映射到外层 x 和 y 轴，岩石层和深度映射到内层 x 和 y 轴。

3.3 数据可视化技术的比较

以上简单介绍了目前比较通用的一些多维数据可视化技术的基本原理，并列举了一系列的实例以确保读者更加直观的对这些技术有较为深刻的认识。接下来将对现有的技术进行比较，希望更加深入的探讨各种技术的优势与不足，从而在今后的研究中改进已有的技术创造出新的技术，以提高可视化在应用中的效率。

表 3.1 是对可视化技术的较为初步的比较，这些比较是以以下一些因素为基础的：

(1) 数据特性：如数据的维度（或者属性数量）或者数据对象的数量等；

(2) 任务特性：如聚类、分类以及多变量热点等；

(3) 可视化特性：如可视重叠以及学习曲线技术等[6]。

表 3.1 多维数据可视化技术的比较

Table 3.1 A attempt at comparing multidimensional data visualization techniques

		聚类	多变量热点	变量数量	数据数量	直接反映数据	可视重叠	学习曲线
几何转化显示技术	散点图矩阵	++	++	+	+	−	O	++
	解剖视图	+	+	-	O	O	+	+
	平行坐标	O	++	++	−	O	−−	O
图标显示技术	脸谱图	O	−	++	+	−	+	−
	星形图	O	−	++	+	−	+	−
密集象素显示技术	递归模式	+	+	++	++	−	++	+
	圆周分段	+	+	++	++	−	++	+
层次技术	层叠式	+	+	O	O	++	O	O

表 3.1 中的“++”表示非常好，“+”表示好，“O”表示一般，“−”表示差，“−−”表示非常差。

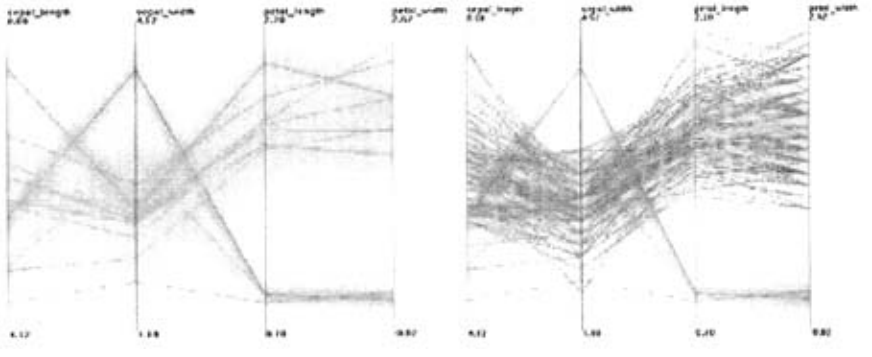
4 多维数据可视化中交互技术的研究

4.1 数据可视化中的交互与变形技术

除了数据可视化技术，对于有效的数据研究还需要一些交互和变形技术[3]。交互和变形技术可以使数据分析人员直接和视图交互，并且按照研究对象动态的改变视图。用户根据领域知识和主观判断利用交互变形技术可以使视图以不同的效果显示出来，从不同的角度对数据进行分析观察，达到很好的数据分析效果。不同的数据可视化方法，对视图的交互和变形技术也有所不同，如上面介绍的各个数据可视化方法，都有各自的可视化技术供用户在与数据视图进行交互时使用。

4.1.1 刷技术(Brushing Technique)

刷是一种突显数据子集的数据可视化技术。主要用于平行坐标中，突显一部分折线而使其他折线不明显,这样使用户更清晰地了解局部数据的变化规律，着重分析用户所关注的部分。刷可以通过不同的方式来实现，可分为基于普通平行坐标的刷和基于分层平行坐标的刷。



(a) (b)

图 4.1 采用分层平行坐标刷技术实现数据可视化。对于同一组数据，(a)进行较为概括的显示(b)进行较为详细程度显示数据

Fig. 4.1 The visualization applying the brushing technique. There is a set of date (a) display it in a general way while (b) in a particular way

基于普通平行坐标的刷，目前的平行坐标可视化系统多数都可以实现此功能，而且采取的方式都各有不同，在 parvis 平行坐标可视化系统主要采用基于角度的刷技术[30]，即根据两相邻坐标轴间线段的斜率范围来确定需要刷的数据。斜率在两条线段夹角范围内的数据将被刷出来。基于角度的刷之后，没有被刷到的数据将变的不明显，被刷到的数据将突显出来，供用户分析研究。在 XmdvTool 数据可视化系统中[31,32]，采用两种方式进行刷操作。一种是控制阴影覆盖的范围的方法来进行刷，一个多维数据的每一维数据如果都在阴影范围之内则被刷出来；另一种方式是通过鼠标扫过的某两相邻坐标间的线段来控制刷的范围，扫过的数据点将被刷出来。分层平行坐标的刷主要是采用基于结构的刷技术(Structure-Based Brush)，被刷出的数据可以用不同的详细程度来显示出来，如图 4.1 所示。

4.1.2 上卷下钻(Drill across & drill through)

在上一章中数据模型的概念中提到过上卷下钻操作。通过上卷和下钻操作可以使数据呈现出不同的详细程度，从而使我们可以从不同的层次上观察和分析数据。图 4.2 中的两图是同一组数据在采用不同粒度子平行坐标中显示的结果。上卷和下钻操作在分层操作中，选择的范围不同时，显示的详细程度也不同。当数据由较为概括的层次向较为细致的层次转换时，用到了下钻操作，反之，则使用上卷操作。

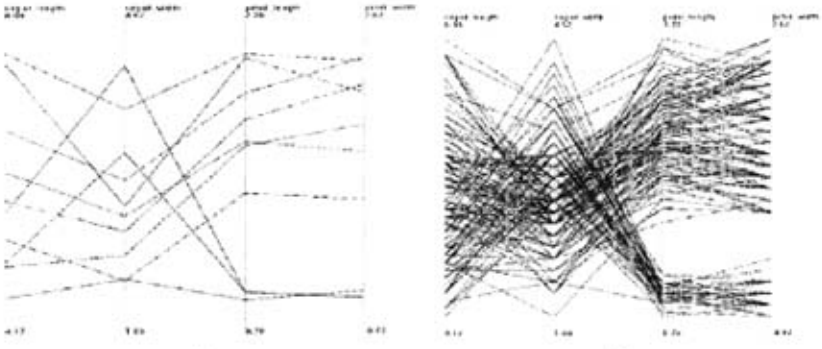


图 4.2 在平行坐标中的上卷和下钻操作。（a）为上卷操作（b）为下钻操作

Fig. 4.2 Operation of drill-across and drill-through in parallel coordinate. (a) is drill-across and (b) is drill-through

由于上卷下钻操作是数据仓库和数据挖掘的基础，因此在各种知识发现数据挖掘系统的实践中，此种交互技术非常广泛的采用，用户可以通过控制在数据层次中的观察点的方式来进行上卷下钻操作。

4.1.3 聚类(Clustering)

聚类就是按照事物间的相似性进行区分和分类的过程，在这一过程中没有教师指导，因此是一种无监督分类。聚类分析则是用数学方法研究和处理所给对象的分类以及各类之间的亲疏程度，是在对数据不作任何假设的条件下进行分析的工具。由聚类生成的簇是一组数据对象的集合，这些对象与同一个簇中的对象彼此相似，与其它簇中的对象相异[33,34]。

在商务上，聚类能帮助市场分析人员从客户基本库中发现不同的客户群，并且用购买模式来刻画不同的客户群的特征。在生物学上，聚类能用于推导植物和动物的分类，对基因进行分类，获得对种群中固有结构的认识。聚类在地球观测数据库中相似地区的确定，汽车保险单持有者的分组，及根据房子的类型、价值和地理位置对一个城市中房屋的分组上也可以发挥作用。聚类也能用于对 Web 上的文档进行分类，以发现信息。作为一个数据挖掘的功能，聚类分析能作为一个独立的工具来获得数据分布的情况，观

察每个簇的特点，集中对特定的某些簇作进一步的分析。此外，聚类分析可以作为其它算法的预处理步骤，这些算法再在生成的簇上进行处理。

作为统计学的一个分支，聚类分析已经被广泛地研究了许多年，主要集中在基于距离的聚类分析。基于 K-means（K-平均值）、K-medoids（K-中心点）和其它一些方法的聚类分析工具已经被加入到许多统计分析软件包或系统中，例如，S-Plus，SPSS 以及 SAS。在机器学习领域，聚类是无指导学习（unsupervised learning）的一个例子。与分类不同，聚类和无指导学习不依赖预先定义的类和带类标号的训练实例。由于这个原因，聚类是观察式学习，而不是示例式学习。在概念聚类（conceptual clustering）中，一组对象只有当它们可以被一个概念描述时才形成一个簇。这不同于基于几何距离来度量相似度的传统聚类。概念聚类由两部分组成：发现合适的簇；形成对每个簇的描述。在这里，追求较高类内相似度和较低类间相似度的指导原则仍然适用。

在数据挖掘领域，研究工作已经集中在为大型数据库的有效和实际的聚类分析寻找适当的方法。活跃的研究主题集中在聚类方法的可伸缩性，方法对聚类复杂形状和类型的数据的有效性，高维聚类分析技术，以及针对大型数据库中混合数值和分类数据的聚类方法。

聚类是一个富有挑战性的研究领域，它的潜在应用提出了各自特殊的要求。数据挖掘对聚类的典型要求如下：

（1）可伸缩性：许多聚类算法在小于 200 个数据对象的小数据集合上工作得很好；但是，一个大规模数据库可能包含几百万个对象，在这样的大规模数据集合样本上进行聚类可能会导致有偏差的结果。我们需要具有高度可伸缩性的聚类算法。

（2）处理不同类型属性的能力：许多算法被设计用来聚类数值类型的数据。但是，应用可能要求聚类其它类型的数据，或二元类型，分类/标称类型，序数型数据，或者这些数据类型的混合。

（3）发现任意形状的聚类：许多聚类算法基于欧几里得距离或者曼哈坦距离度量来决定聚类。基于这样的距离度量算法趋向于发现具有相近尺度和密度的球状簇。但是，一个簇中可能是任意形状的。提出能发现任意形状簇的算法是很重要的。

（4）用于决定输入参数的领域知识最小化：许多聚类算法在聚类分析中要求用户输入一定的参数，例如希望产生簇的数目。聚类结果对于输入参数十分敏感，参数通常很难确定，特别是对于包含高维对象的数据集来说，更是如此。要求用户输入参数不仅加重了用户的负担，也使得聚类的质量难以控制。

（5）处理噪声数据能力：绝大多数现实世界中的数据库都包含了孤立点，空缺，未知数据或者错误的数据。某些聚类算法对于这样的数据敏感，可能导致低质量的聚类结果。

（6）对于输入记录的顺序不敏感：某些聚类算法对于输入数据的顺序是敏感的。例如，同一个数据集合，当以不同的顺序提交给同一个算法时，可能生成差别很大的聚类结果。开发对数据输入顺序不敏感的算法具有重要的意义。

（7）高维性：一个数据库或者数据仓库可能包含若干维或者属性。许多聚类算法擅长处理低维的数据，可能涉及两到三维。人类最多在三维的情况下能够很好地判断聚类的质量。在高维空间中聚类数据对象是难度较大的，特别是考虑到这样的数据可能非常稀疏，而且高度倾斜。

（8）基于约束的聚类：现实世界的应用可能需要在各种约束条件下进行聚类，假设你的工作是在一个城市中为给定数目的自动提款机选择安放位置。为了做出决定，你可以对住宅区进行聚类，同时考虑如城市的河流和公路网，每个地区的客户要求等情况。要找到能满足特定的约束，又具有良好聚类特性的数据分组是一项高要求的任务。

（9）可解释性和可用性：用户希望聚类结果是可解释的，可理解的和可用的。也就是说，聚类可能需要和特定的语义解释和应用相联系。应用目标如何影响聚类方法的选择也是一个重要的研究课题。

目前在文献中存在大量的聚类算法。算法的选择取决于数据的类型、聚类的目的和应用。如果聚类分析被用作描述或探查的工具，可以对同样的数据尝试多种算法，以发现数据可能揭示的结果。

4.1.4 维度的显示控制(Controlling of dimensions)

在数据的显示过程中，为了方便用户进行观察，常常需要对数据各个维度的显示进行相应的控制，以便使显示更加高效。

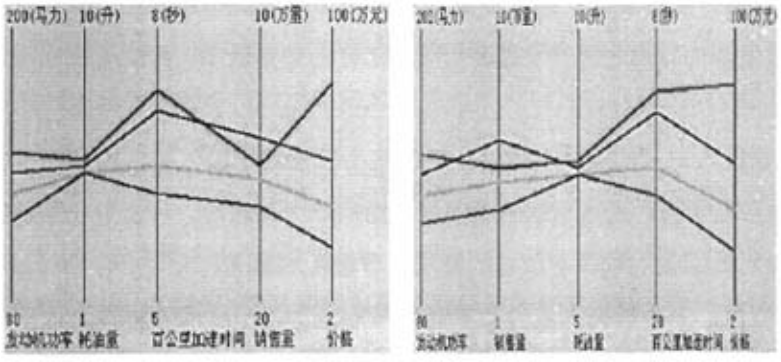


图 4.3 平行坐标中采用维交换的同一组数据的显示效果

Fig. 4.3 The change of dimensions in Parallel coordinate

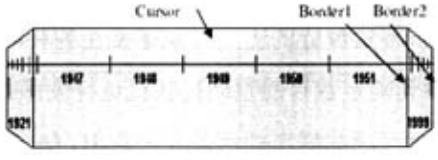


图 4.4 一个关于时间轴的视点控制器

Fig. 4.4 Perspective wall timeline

(1)维数控制：在分析数据的过程中往往会忽略掉一部分不重要的数据，并且去除干扰数据。在平行坐标、基于图标以及密集象素可视化技术均可以通过对数据属性数量的控制，只将用户关心的属性显示出来，这样既减小图形的复杂度，又减少了不重要数据对显示结果的干扰，使用户更容易对数据进行分析。刷技术可以看作对数据的行进行分解，而对维数的控制则可以看作是对列作分解。在实际应用中，用户可以只保留自己感兴趣的属性，观察它们之间的关系，去掉不相关的数据以后更容易发现变化的规律。

(2)维的交换显示：在显示过程中，适当的交换数据可以把用户认为属性关系密切的属性安排在相邻或相近的位置，更好的呈现属性间的关系和规律。在未知属性间的关系时，可以试探的掉换坐标轴次序，以发现不同属性间隐含的关系。对于维交换技术可以在很多可视化方法（如平行坐标，图标技术，密集象素技术等）中采用，如图 4.3 所示。

(3)维放缩：维放大主要应用在需要局部数据放大的情况下，比如在完成刷操作之后，刷出的数据范围往往比较小，这时可以将该区域的数据用全局范围来显示。在数据量大时，这种方法能取得比较好的效果，然而，这样容易失去对全局的把握。一般采用将局部放大的可视化视图与全局的视图结合起来观察，可以避免对数据的片面理解。当数据量小且分散时，采用维缩小可以集中观察数据会更容易观察数据的变化趋势。如图 4.4 所示，是一个针对时间轴的视点控制器的模型，用户可以灵活的对数据进行维的放缩，参见文献[35,36]。对于维度的显示控制，在下面的章节还要进行更深入的研究与讨论。

在前面的章节，已经介绍了一些较为普及的数据可视技术与可视化交互技术。其中多维数据维度的控制对于数据可视化的效果有着比较重要的意义。采用合理的算法对维度的数量以及维度的排列做出规划，既可以降低一些无用的属性对于数据挖掘结果的影响还可以方便用户观察具有密切关系的属性之间的关系。

4.2 维度控制的意义

前文提到，在数据对于维度的控制，当用户明确自己所关心的属性以及属性之间的关系时，可以直接对数据的维进行删减或者排列；而在未知属性间关系或者属性与主题的相关性时，一般采用试探性掉换显示顺序的方法与可视化进行交互控制。试探性的掉换显示顺序的做法，在数据维数众多时显然效率比较低下，用户很难找到一个最为理想的排列顺序。在没有较为合理的部署的情况下，有时用户可能忽略一些看似与主题不相关的规律，从而从一些程度上会制约可视化的显示效果，参见图 4.5。

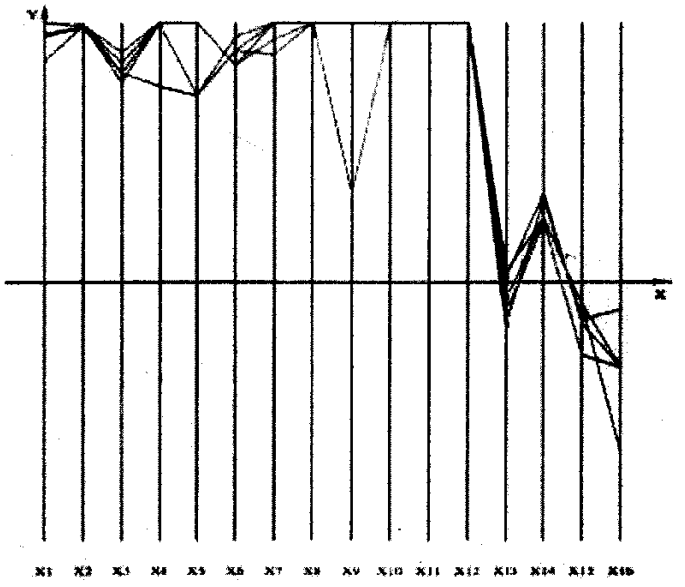


图 4.5 一种产品有最大的利润而在缺陷属性 x3~x6 却没有取到最小值

Fig. 4.5 The batches with the highest yield and do not have the lowest defects in x3~x6

图 4.5 中，平行坐标显示了某企业的产品所获利润与产品质量相关的一组数据。在图中属性 x1 表示产品的利润，x2 是产品质量的一组度量，x3～x12 表示产品存在的一些缺陷的度量，x13～x16 是产品的一些物理特性。按照一般常识判断： x1 取最大值，应该是 x3～x12 最小，因为产品销路好质量相应的也应该好。有趣的是，通过实际观察结果却不是这样，获得利润最大的产品恰恰在 x3～x6 属性中存在一定的缺陷[35]。对于这样的结果，如果数据的维数很大彼此间关系又明晰的情况下，采用试探性的调整维度显示顺序很难发现这些有趣的规律。因此，需要我们采用一些数学手段定量的研究数据维之间的一些关系，去掉一些不相关的维，并调整一些维的排列顺序对数据可视化显示具有很深刻的意义。

4.3 基于相似度的维排列算法

4.3.1 相似算法的基础

在数据挖掘中，常常应用聚类分析的方法对数据进行观察式学习（或称无训练例学习）。在聚类的数据集合包含 n 个数据对象，许多基于内存的聚类算法选择如下两种代表性数据结构：

（1）数据矩阵（或称对象与变量结构）：它用 p 个变量（或称度量、属性）来表现 n 个对象。这种数据结构是关系表的形式，或可以看成是 n×p 的矩阵。

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{n1} & \cdots & x_{ni} & \cdots & x_{np} \end{bmatrix} \tag{4.1}$$

（2）相似度矩阵（或称对象与对象结构）：存储 n 个对象两两间的近似性，表现形式是一个 n×n 维的矩阵。

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & & 0 & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix} \tag{4.2}$$

在这里 d(i, j)是记录 $r_i(x_{i1},\ldots,x_{if},\ldots,x_{ip})$和 $r_j(x_{j1},\ldots,x_{jf},\ldots x_{jp})$之间相异性的量化表示，由于 d(i, j)与 d(j, i)表示意义相同，因此只写下三角阵，当对象 i 和 j 越相似其值越接近 0，反之则 d(i, j)的值越大。通过数据的相似度的计算可以实现海量数据的区分、分类以及聚类的分析。对于对象间的相似度（或差异度）最简单的计算方法就是基于对象间的距离来计算的。

距离函数为：

$$d(i,j)=\sqrt[q]{w_1\left|x_{i1}-x_{j1}\right|^q+w_2\left|x_{i2}-x_{j2}\right|^q+\cdots+w_p\left|x_{ip}-x_{jp}\right|^q} \tag{4.3}$$

1）当 $w_i=1$（其中 i=1, 2, …, p）且 q=1 时，为曼哈坦距离。

2）当 $w_i=1$（其中 i=1, 2, …, p）且 q=2 时，为欧几里德距离。

3）当 $w_i=1$（其中 i=1, 2, …, p）且 q≥1 时，为名考斯基距离。

4）当 w_i 为任意值且 q=2 时，为加权欧几里德距离。

相似度算法不仅可以应用于数据挖掘过程中数据间的聚类分类研究，还可以进一步扩展到平行坐标系各维度显示控制方面。针对平行坐标系中的数据在显示过程中缺乏合理组织影响显示效果的问题，可以采用基于相似度的方法来对各个维度的相似性进行定量的比较，再利用优化算法得到最佳的组合方案来指导可视化显示。

4.3.2 基于相似度的维排列算法的描述

在数据挖掘过程中，数据的相似度从二维表的结构上考虑，可以理解为数据矩阵中各个行向量间距离的度量。若把二维表的各属性列也看作向量，如式(4.4)所示，同样采用(4.3)算法便可以得到各个维之间的相似度。

$$V_j=(x_{1j},x_{2j},\cdots,x_{ij},\cdots x_{nj})^T \tag{4.4}$$

按照（4.6）式的表示，可以按各种不同需求利用不同的距离函数计算各向量间的距离，进而得出各维度之间的相似关系。由于各向量 v_j 之间表示的内容意义各不相同，因此在单位和取值上存在很大差异。不能直接对各向量进行比较，必须对向量中的每一个分量进行标准化处理，以避免数据本身的差异对显示结果的影响。维向量的标准化也可以采用如（4.5）式的方法进行。

同各数据向量的计算方法相似，维向量也采用向量间距离法来计算相似行，即当两个维向量间距离越小表示向量越相似，反之则越不相似。常见的相似度算法比较多，主要分为全局相似度和局部相似度，基本可以满足用户的各种需求。一般，采用不同的标准化方法便可以得到不同的相似度结果。

（1）全局相似度算法：全局相似度算法是将全部数据作为研究对象，将全部数据在某一维度上的属性分量组成向量进行计算。这样所有数据内部隐含的规律可以定量的进行表示。

$$S_{trans}(V_k,V_l)=\sqrt{\sum_{i=0}^{N-1}((a_{ik}-mean(V_k))-(a_{il}-mean(V_l)))^2} \quad (4.5)$$

其中 $mean(V_i)=\frac{1}{N}\sum_{k=0}^{N-1}a_{ik}$。

采用(4.5)算法计算两个数据维之间的相似度，适用于数据记录各个属性上值标准差比较小的情况；若标准差比较大，可能使 $S(V_k,V_l)$受到个别值的影响。

（2）基于变量缩放比例的相似度算法：此种算法描述数据集中数据规模的相似程度的规律，具体算法描述如下：

$$S_{scaling}(V_k,V_l)=\sqrt{\sum_{i=0}^{N-1}(b_{ik}-b_{il})^2} \quad (4.6)$$

其中 $b_{ij}=\frac{a_{ij}-MIN(V_j)}{MAX(V_j)-MIN(V_j)}$。

该算法对维向量的取值进行规范化处理，使各个维向量的分量的取值在相同的取值范围上。b_{ij} 是对 a_{ij} 进行规范化处理之后的结果。

（3）局部相似度算法：在实际应用中，数据量非常大，追求数据在全局上的相似趋势一般不如追求数据在一定范围内的局部相似，可以采用如下算法描述：

$$S_{sync}(A_k,A_l)=\sqrt{\sum_{z=i}^{j}(b_{zk}-b_{zl})^2} \quad (4.7)$$

其中 b_{zk} 与 b_{zl} 分别是对 a_{zk} 和 a_{zl} 进行规范化处理的结果，而 i,j 的取值满足：$0\leqslant i<j<N$。

4.3.3 维度的排序与过滤

根据前文的算法得出记录集中各个维之间的相似度，并写成相似度矩阵的形式。用户可以根多维数据的各个维度的相似性而对可视化过程中维度排列进行规划，并且可以

度一些与主题关联比较小的属性进行过滤。(4.8)式采用上述相似算法求得的各 S 值组成一个 k×k 阶矩阵如下（其中 $S(A_i,A_j)$ 表示维度 i 与维度 j 的相似度）：

$$S=\begin{bmatrix} S(A_0,A_0) & \cdots & S(A_{k-1},A_0) \\ \vdots & \ddots & \vdots \\ S(A_0,A_{k-1}) & \cdots & S(A_{k-1},A_{k-1}) \end{bmatrix} \tag{4.8}$$

其中：$S(A_i,A_j)=S(A_j,A_i),\forall i,j=0,\cdots,(k-1)$都有$S(A_i,A_i)=0$。

（1）维度排列的规划：

根据用户的需要可以对维度进行整体规划，即将所有数据的所有的维度作为研究对象，对其进行规划得到一个整体收益比较高的维度排列最终用于显示数据。(4.8)中获得了记录集中每两个维之间的相似度并组成相似度矩阵；为了便于描述再定义一个相邻矩阵，来表示两个维度之间是否相邻，如(4.9)所示：

$$N=\begin{bmatrix} n_{00} & \cdots & n_{(d-1)0} \\ \vdots & \ddots & \vdots \\ n_{0(k-1)} & \cdots & n_{(k-1)(k-1)} \end{bmatrix} \tag{4.9}$$

N 为相邻矩阵，对于任意的 i 与 j 都存在：$n_{ij}=n_{ji}$ 并且 $n_{ii}=0$。而 n_{ij} 的取值如下：

$$n_{ij}=\begin{cases} 1 & \text{当i和j相邻} \\ 0 & \text{否则} \end{cases}$$

在定义相似矩阵及相邻矩阵之后，参考维排列方案定义如下：

$$\text{income}=\sum_{i=0}^{k-1}\sum_{j=0}^{k-1} n_{ij}S(A_i,A_j) \tag{4.10}$$

如（4.10）所示，当 income 取到最小值时，所得到的维排列规划各个维度之间整体差异度较小，也就是各个维之间具有比较大的相似度。而当 S 一定时，相邻矩阵的选取是使 income 获得最小值的关键。在可视化显示时，各个维两两相邻，因此维相邻矩

阵应该是一个在每一行每一列上都保持只有一个非零数值的矩阵。可以按照数据结构中八皇后问题进行解决，可以参照以下递归算法进行编程：

```
void permutation(char a[], int m, int n)
{
    int i;
    char t;
    if (m<n-1) {
        permutation(a, m+1, n);
        for (i=m+1; i<n; i++) {
            t=a[m]; a[m]=a[i]; a[i]=t;
          permutation(a, m+1, n);
          t=a[m]; a[m]=a[i]; a[i]=t;
            }
    } else
        {
          printf("%s ", a);
        }
}
```

采用此算法对维度的排列进行规划能找出趋势较为一致的维度并将它们安排在比较接近的位置，这样用户就能比较容易的发现比较隐秘的一些规律和模式。

（2）维度的排序与过滤：

在数据可视化过程中，也可以针对某一维度来规划与其相关的数据进行局部维度的排列。这样，就可以得到与某一主题相关的一种可视化规划。根据式(4.8)的相似度，其中每一行（或一列）都是某一属性对其他属性的相似度。我们可以将该属性作为一个主题进行研究。对该行（或列）的相似度值进行排序，再将相似度所对应的维度按照排列的顺序进行可视化显示，使相似度较为接近的数据排列在比较接近的位置，从而便于

用户进行观察。并且，对于一些与主题维度差异比较大的维度可以按照用户给定的进行过滤， 从而进一步提高显示的效率。

4.3.4 数据规范化

在数据挖掘预处理阶段有一个环节是数据规范化。该环节将属性数据按比例缩放，使之落入一个小的特定区间。对数据挖掘中的分类算法，如涉及神经网络的算法或诸如最临近分类和聚类的距离度量分类算法，规范化特别有用；不仅如此，在结果可视化环节对于记录属性的规范化同样可以防止某些具有初始值较大的属性相对较小的属性权重较大的情况发生。在维相似度算法中，数据规范化同样比较重要，由于各维度之间所记录的属性各不相同，因此各属性之间的单位也可能不同，因此对数据进行规范化可以使数据落在一个较小的范围，既可以便于显示，又有效的控制属性初始值的权重对结果的影响。

目前有多种数据规范化方法，较为常见的有：最小-最大规范化、z-score 规范化和按小数定标规范化等。

(1) 最小-最大规范化：

$$b_{ij}=\frac{a_{ij}-MIN(V_j)}{MAX(V_j)-MIN(V_j)} \tag{4.11}$$

最小-最大规范化对原始数据进行线性变换。(4.11)中的 MIN（V_j）和 MAX（V_j）分别表示维向量 V_j 中所有数据的最小值和最大值。a_{ij} 表示数据矩阵中第 i 行第 j 列的数据。使用(4.7)式的方法对数据进行标准化可以使数据按照值域进行标准化，使数据独立于维向量的取值范围，使其取值在区间[1,-1]之间。

(2) z-score 规范化（或零-均值规范化）：

$$EA_i=\frac{1}{n}(|x_{1i}-m_i|+|x_{2i}-m_i|+\cdots+|x_{ni}-m_i|) \tag{4.12}$$

其中 m_i 是第 i 组数据的平均值，即

$$m_k=\frac{1}{n}(x_{1k}+x_{2k}+\cdots+x_{nk})$$

$$z_{ik} = \frac{x_{ik} - m_k}{EA_i} \tag{4.13}$$

在数据记录集中各个属性最大和最小值未知，或孤立点左右了最大-最小规范化时，可以采用 z-score 规范化。

(3) 小数定标规范化：

通过移动属性 A_i 的小数点的位置进行规范化。小数点的移动位数依赖于 A_i 的最大绝对值。A_i 的的值 v_i 被规范化为 s_i，由下式计算：

$$s_i = \frac{v_i}{10^j} \tag{4.14}$$

其中，j 是使得 $\mathrm{Max}(|s_i|)<1$ 的最小整数。

经过数据规范化使数据变成无单位数据，再通过距离公式可以方便的求得相似度，进行数据挖掘的后续操作，很大程度上可以降低了挖掘过程中孤立点的影响，为依据相似度算法排列维度做好准备。

4.3.5 实验结果分析

选取一组篮球运动员的技术指标对基于维相似度排序算法进行实验，这些数据包括运动员的投篮、罚篮、三分球、扣篮、防守篮板、进攻篮板、盖帽、抢断、传球、进攻意识、防守意识、速度、敏捷、弹跳、控球、力量、耐力、爆发力以及内线得分等 19 个维度的指标。对这些数据按照录入顺序进行可视化显示的效果如图 4.7 所示。

按照(4.5)的相似算法对各个维之间的相似度进行计算 S_{trans}，得到相似矩阵 S 如图 4.6 所示。再按照球员扣篮指标为观察主题，提取其他维度与它的相似度进行排序，最终得到数据的可视化显示结果如图 4.8 所示。

凭借篮球领域的相关经验和常识可以得到：篮球运动员的扣篮能力一般由运动员的弹跳、力量、爆发力直接决定。运动员扣篮能力又与其内线得分能力、进攻和防守能力有着直接的联系。另外，运动员的扣篮能力一般与其三分球、罚篮以及传球等纯技术方面的指标往往关联不大。

	FieldGoal	FreeThrow	ThreeGoal	Dunk	DefRe	OffRe	Steal	Block	Pass	OffAware	DefAware	Speed	Quickness	Jump	CtrBall	Strength	Durability	Energy	InsideGoal
FieldGoal	0	394.48	763.35	753.37	612.39	908.57	555.70	1078.6	707.64	327.63	402.64	319.08	327.13	460.12	365.17	544.84	425.98	469.24	624.60
FreeThrow	394.48	0	978.40	869.56	657.26	1038.1	689.94	1256.1	918.58	529.57	354.37	370.72	595.59	531.31	571.62	397.70	682.87	965.73	537.44
ThreeGoal	763.35	978.40	0	1102.7	1098.9	1180.9	710.97	1216.3	665.79	886.03	912.69	839.74	838.49	851.70	637.18	1035.2	1000.5	813.71	981.59
Dunk	753.37	869.56	1102.7	0	602.16	694.41	838.41	832.28	1037.3	744.30	704.56	809.95	809.12	547.12	858.54	618.93	815.17	753.11	569.19
DefRe	612.39	657.26	1098.9	602.16	0	552.80	734.58	795.40	968.18	809.38	529.58	660.72	662.59	558.95	760.31	357.37	605.34	692.72	369.17
OffRe	908.57	1038.1	1180.9	694.41	552.80	0	891.46	542.87	1002.8	922.52	784.79	974.07	970.29	769.38	982.26	636.77	934.59	847.83	648.39
Steal	555.70	689.94	710.97	838.41	734.58	891.46	0	1022.8	611.37	644.19	569.74	526.64	518.49	550.41	453.40	680.82	691.78	656.75	651.35
Block	1078.6	1256.1	1216.3	832.28	795.40	542.87	1022.8	0	1067.3	1116.5	950.80	1152.7	1145.6	935.39	1130.9	880.67	1151.5	985.39	855.88
Pass	707.64	918.58	665.79	1037.3	968.18	1002.8	611.37	1067.3	0	803.60	729.06	724.22	717.08	761.74	542.66	919.56	874.51	869.98	827.13
OffAware	327.63	529.57	886.03	744.30	809.38	922.52	644.19	1116.5	803.60	0	452.89	377.43	382.11	493.77	475.39	564.10	450.66	502.72	445.90
DefAware	402.64	354.37	912.69	704.56	529.58	784.79	569.74	950.80	729.06	452.89	0	420.28	414.59	462.60	516.31	520.19	459.44	483.19	397.37
Speed	319.08	370.72	839.74	809.95	660.72	974.07	526.64	1152.7	724.22	377.43	420.28	0	131.89	442.69	327.49	581.68	395.84	573.62	499.67
Quickness	327.13	595.59	838.49	809.12	662.59	970.29	518.49	1145.6	717.08	382.11	414.59	131.89	0	442.58	335.57	585.58	401.54	567.09	495.84
Jump	460.12	531.31	851.70	547.12	558.95	769.38	550.41	935.39	761.74	493.77	462.60	442.69	442.58	0	504.84	512.36	560.60	563.45	704.97
CtrBall	365.17	571.62	637.18	858.54	760.31	982.26	453.40	1130.9	542.66	475.39	516.31	327.49	335.57	504.84	0	682.62	556.42	549.09	598.25
Strength	544.84	397.70	1035.2	618.93	357.37	636.77	680.82	880.67	919.56	564.10	520.19	581.68	585.58	512.36	682.62	0	539.20	709.07	354.00
Durability	425.98	682.87	1000.5	815.17	605.34	934.59	691.78	1151.5	874.51	450.66	459.44	395.84	401.54	560.60	556.42	539.20	0	641.29	473.01
Energy	469.24	965.73	813.71	753.11	692.72	847.83	656.75	985.39	869.98	502.72	483.19	573.62	567.09	563.45	549.09	709.07	641.29	0	540.85
InsideGoal	624.60	537.44	981.59	569.19	369.17	648.39	651.35	855.88	827.13	445.90	397.37	499.67	495.84	704.97	598.25	354.00	473.01	540.85	0

图 4.6 实验数据的相似矩阵

Fig. 4.6 The Similarity Matrix of Experiment Data

从系统可视化的结果上来看，运动员的弹跳、内线得分、防守篮板、进攻篮板、力量等指标都排在了较为接近扣篮指标的位置，而罚篮、传球以及三分球等指标都排列在较远的位置。这样用户便可以很方便的对数据之间的关系进行观察与分析。

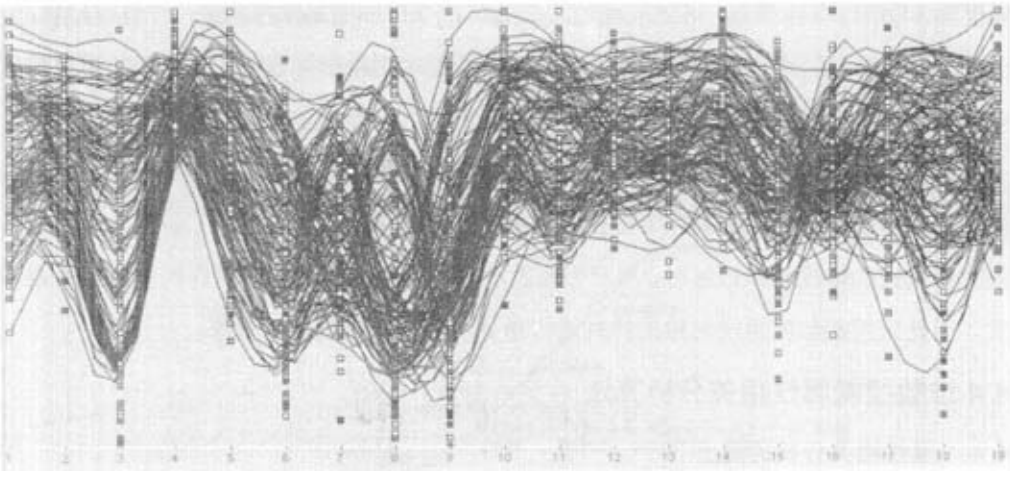


图 4.7 未经过维排序处理的数据可视化效果

Fig. 4.7 The Visualization of Dataset without Dimensions Disposing

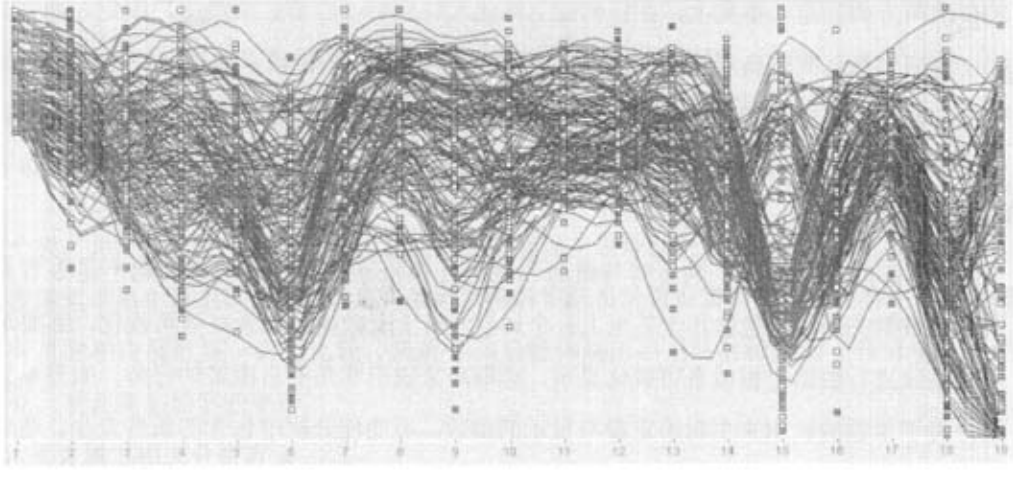


图 4.8 经过维排序处理的数据可视化效果

Fig.4.8 The Visualization of Dataset without Dimensions Disposing

实验中所采用的数据是一些具有很强扣篮能力的运动员的信息，通过观察图 4.8 中的可视化结果，得到一些结论：此类运动员具有比较强的弹跳力、力量以及内线得分能力，而和篮板球能力却不高。通过对距离扣篮指标较远的三分球(x 轴坐标 19)和盖帽(x

轴坐标 15)指标进行观察，可以发现运动员的这两项指标明显分成两支，一直指标较高，而另一支较低，很难分析出扣篮能力强的运动员与这两项指标直接关系。因此可以将一些不相关的维度信息放在比较靠后的位置次要考虑，甚至可以将它们过滤掉。

从实验可以看出，当数据的维度很高的情况下，如果数据维度的显示没有合理的排列图形的复杂度会大大提高，图形相互的叠加也会严重影响显示效果。并且，当有隐含规律和模式的维度相距很远时，用户很难发现一些看似没有关系的属性间隐藏的有趣模式。因此采用相似度算法对维的排列进行指导可以提高可视化的效果。

4.4 维数控制属性相关分析算法

4.4.1 属性相关分析的提出

在对数据仓库、OLAP 工具以及可视化工具中的多维数据分析时，往往缺乏自动的概化机制，用户必须显式的告诉系统，哪些应当包含在类分析中，每个维应概化到更高的层次两方面信息。事实上，在任何维上概化和特化每一步都必须由用户制定。通常，对于维应当概化到较高层次并不困难。如图 4-2 中，location 维，我们即可指定其概化的 country 层次。即使用户没有给出显式的声明系统也可以设置一个缺省的阈值由 1 到 5，维可以概化到不同的层次。如果用户对于系统给定的概化层次不满意，可以上卷下钻到需要的维。

然而，对用户来说，确定哪些维应当包含在类特征分析及可视化中则不是很容易[37-39]。数据关系通常包含几十甚至上百个属性，对于有效挖掘以及高效可视化，需要选择那些维进行数据挖掘或者可视化显示，对用户来说确实是非常困难的任务。如果单凭用户对于主题的认识来主观给定参与显示的维数，可能使分析中包含的属性太少，造成挖掘的描述结果不完全；也可能包含太多属性，降低了系统的挖掘与显示效率。

在数据挖掘的概念描述环节经常会用到可视化方法对系统中的数据或者数据挖掘的中间结果进行显示，为了降低不相关维对于挖掘以及可视化工作的干扰，引进一种算法对属性进行相关性分析，以过滤掉统计上不相关或弱相关的属性，而保留对挖掘或显示任务最相关的属性。在数据挖掘中，通常把包含属性或维相关分析的类特征化称为解析特征化(analytical characterization)，把包含这种分析的类比较称为解析比较(analytical

comparison)[1]。更直观地如图 4.7 所示，对于一个给定的类，如果该属性或维的值可能用于区分该类与其他类，则该属性或者维被认为是与概念高度相关的[41,42]。

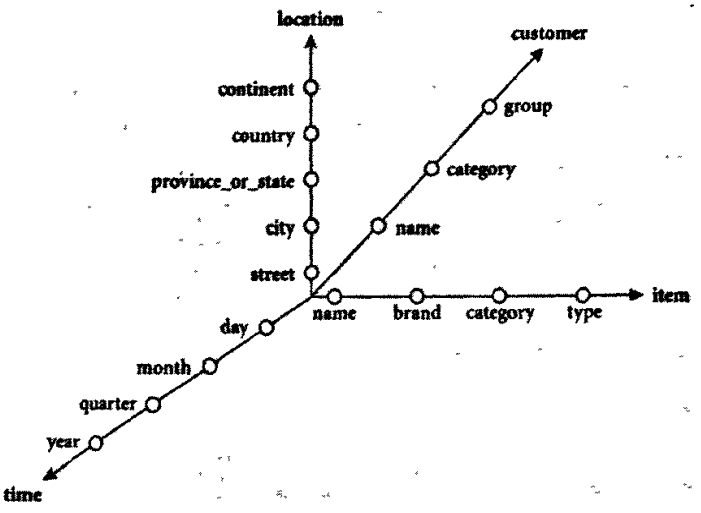


图 4.7 某商务查询模型：一个星形网模型

Fig. 4.7 A business query model: A Star net model

同样的工作也可以用于多维数据的可视化中，可以将一组某一主题数据集看作是一个类，并对数据集中的每一个属性进行区分其他主题数据集的能力的评估。如：在分析篮球运动员中后卫队员的属性时，就可以将后卫队员的数据集的各个属性与中锋、前锋队员的相应属性逐一进行比较，最终找到最能体现出后卫队员特性的一些维用于可视化，从而提高显示效率。

4.4.2 属性相关分析方法

属性相关性分析的基本思想是计算一种用于量化属性与给定概念与所要反映主题相关性的度量，这种度量一般是信息增益、不确定性和相关系数[43,44]。其中信息增益分析技术目前已得到广泛的应用，它可以与基于多维数据分析的方法集成在一起，删除信息量较少的属性，收集信息量较多的属性，用于数据的挖掘与可视化。

设 S 是一个训练样本的集合，其中每个样本的类标号是已知的。事实上，每个样本是一个元组，一个属性用于确定训练样本的类。假定有 m 个类。设 S 包含 s_i 个 C_i 类样

本，i=1，…，m。一个任意样本属于类 C_i 的可能性是 s_i/s，其中 s 是集合 S 中对象的总数。对于给定样本分类所需的期望信息是

$$I(s_1,s_2\cdots s_m)=-\sum_{i=1}^{m}\frac{s_i}{s}\log_2\frac{s_i}{s} \tag{4.16}$$

上面公式（4.16）是一个信息熵公式，熵是香农(Shannon)理论的特征性概念，它与信息内容的不确定程度有等价关系，与热力学上的熵没有关系。其中具有值$\{a_1,a_2,\ldots,a_k\}$的属性 A 可以用来将 S 划分维子集$\{S_1,S_2,\ldots,S_k\}$，S_j 包含 S 中 A 值为 a_j 的那些样本。设 S_j 包含类 C_i 的 s_{ij} 个样本。它的加权平均为：

$$E(A)=\sum_{j=1}^{k}\frac{s_{1j}+\cdots s_{mj}}{s}I(s_{1j}+\cdots+s_{mj}) \tag{4.17}$$

这样，在属性 A 上获得的信息增益定义为

$$Gain(A)=I(s_1,s_2,\cdots,s_m)-E(A) \tag{4.18}$$

在这个相关性分析方法中，利用 S 可以计算出每一个属性的信息增益度。具有较高信息增益的属性是给定集合中具有最高区分度的属性，应当给予重视[1]。通过信息增益的属性的评定。这种评定可以用作相关分析，当然也可以指导可视化过程中的维数的显示以及维排列。为了防止忽略一些信息还应结合(4.2)中的相似度算法对于维度的显示顺序进行控制。

4.4.3 属性相关分析步骤描述

以上对于控制可视化维数的属性相关性分析方法进行了阐述，现在将对数据属性相关分析的步骤进行描述，以便在系统的开发中应用此种算法。具体步骤如下：

(1) 数据收集：通过查询处理，收集目标类和对比类的数据。对于类比较或者概念描述，目标类和对比类都由用户在数据挖掘查询中提供。对于类的特征化，目标类是要特征化的类，而对比类是不在目标类中的可比较数据，从某种意义上讲，目标类是一种训练例。

(2) 使用保守的 AOI(Attribute Oriented Induction)进行预相关分析：此步骤识别属性和维的集合，选择的相关性度量来评价它们。由于维的不同层次对于给定的类具有很多

不相同的相关性，原则上，定义维概念层的每个属性都应当包含在相关性分析中。然而有一些属性很显然对数据挖掘没有很大的意义，因此可以直接删除或概化这些与挖掘无关的具有大量不同值的属性（如 name，phone）。这种方法可以不用利用算法便可以直接实现[42]。对于可视化所描述的数据，这些属性在表示数据概念或者数据间关系多半也没有意义。由于没有采用定量分析而删除或概化一些属性，因此需要保守一些，进行的 AOI 使用的属性分析的阈值要合理的大，使得更多的（但非所有的）属性在选定度量的进一步相关分析中被考虑。这样使用 AOI 得到的关系称为挖掘任务的候选关系。

(3) 使用选定的相关分析度量删除不相关和弱相关的属性：使用选定的相关分析度量，评估候选关系中的每个属性。此步骤所用的相关性度量可以建立在数据挖掘系统中，或由用户提供。可以使用上面介绍的信息增益度量。根据计算属性与数据挖掘任务的相关性，对属性排序。然后删除与概念描述或可视化任务不相关或弱相关的属性。可设置一个阈值来定义“弱相关”。其结果为初始目标类工作关系和初始对比类工作关系。

(4) 使用 AOI 产生概念描述及可视化结果：使用一组不太保守的属性概化阈值进行 AOI。如果类描述任务是类特征化，这里只包含初始目标类工作关系。如果类描述任务是类比较，初始目标类工作关系和初始对比类工作关系都要包含在分析中。

属性归纳过程进行了两次，一次是预相关分析（步骤 2），另一次是在初始工作关系上归纳（步骤 4）。已选定度量进行属性相关分析（步骤 3）所用的统计可以在步骤 2 的数据库扫描时收集。

4.4.4 数据特化实例

在采用可视化进行概念描述时，涉及许多属性，应当在可视化之前采用类似数据挖掘过程中解析泛化技术，先删除不相关或弱相关的属性。以下是一个篮球前锋运动员数据集，在可视化之前需要解析泛化，具体数据如下：

表 4.2 前锋运动员数据（目标类）

Table 4.2 Forward player data (Object class)

School_age	League	Birth_country	Strength	Residence_city	InsideGoal	Count
U	NBA	USA	66...99	New York	Very_good	16
H	NBA	Foreign	66...99	Sydney	Excellent	22
U	ABA	Foreign	66...99	Ottawa	Excellent	18
H	NBA	Foreign	66...99	Ottawa	Excellent	25
U	NBA	USA	33...66	L.A	Excellent	21
H	ABA	USA	33...66	L.A	Excellent	18

表 4.3 后卫运动员数据（对比类）

Table 4.3 Guard player data (Comparing class)

School_age	League	Birth_country	Strength	Residence_city	InsideGoal	Count
U	Science	Foreign	<=33	London	Very_good	18
H	Business	USA	<=33	New York	Fair	20
U	Business	USA	<=33	New York	Fair	22
H	Science	USA	33...66	Chicago	Fair	24
U	Engineering	Foreign	33...66	Paris	Very_good	22
H	Engineering	USA	<=33	L. A	excellent	24

采用解析特征化对研究生类进行维数控制具体步骤如下：

首先，收集目标数据，收集目标数据前锋运动员数据以及对比类数据后卫运动员数据，以便进行相关分析。接下来采用保守的属性概化删除一些对于可视化不相关的数据属性，将诸如 name，nationality，residence_city 删除掉。设 C_1 对应于前锋运动员类，C_2 对应于后卫运动员类。在前锋运动员中有 120 个样本，后卫运动员有 130 个样本。根据公式（4.16）计算出给定样本分类的期望信息：

$$I(s_1,s_2)=I(120,130)=-\frac{120}{250}\log_2\frac{120}{250}-\frac{130}{250}\log_2\frac{130}{250}=0.9988$$

下一步，需要计算每个属性的熵。以 League 为例，对 League 的每个值观察前锋运动员和后卫运动员的分布，并计算期望信息。

League= "NBA"

S_{11}=84　　S_{21}=42　　$I(S_{11},S_{21})$=0.9183

League= "ABA"

S_{12}=36　　S_{22}=46　　$I(S_{12},S_{22})$=0.9892

League= "CBA"

S_{13}=0　　$S_{23=42}$　　$I(S_{13},S_{23})$=0

根据公式(4.17)，如果样本根据 League 划分，则给定的样本进行分类所需的期望信息是：

$$E(League)=\frac{126}{250}I(s_{11},s_{21})+\frac{82}{250}I(s_{12},s_{22})+\frac{42}{250}I(s_{13},s_{23})=0.7873$$

因此，由这样的划分的信息增益是：

$$Gain(League)=I(s_1,s_2)-E(League)=0.2115$$

在数据挖掘系统中，采用该方法进行概念描述或者挖掘类的比较之前的预处理，删除或概化一些不相关的维度信息，以提高数据挖掘系统的准确性与效率。数据可视化技术实际上也是一种概念的描述，只是它将数据挖掘过程中以数值描述概念的描述转换为以图形的方式进行描述[45]。因此采用信息增益法对可视化中的数据维数进行控制也是比较适用的。由于在进行评估的过程中需要有用来比较的训练样本来指导评估，因此可能会给可视化带来一些不方便，但只要用户采取一些统计或查询手段获取一些具有指导意义的可靠的参考数据，相信此种基于属性相关技术的维数控制算法还是比较实用的。

5 篮球运动员技术指标分析系统

5.1 篮球运动员技术指标分析系统介绍

运动员的技术指标分析在运动员的选拔和训练方面指导作用都占有非常重要的地位。通过对运动员的素质和技术指标的分析，可以预测出运动员的未来发展情况，可以区分运动员的体能类型（如速度耐力项群或速度力量项群）[46]，而在对这些问题的分析过程中，经常会遇到多维的数据样本，为了清楚地将众多变量间的相互关系表达和显示出来，可以借助图形在一定程度上可以将抽象和复杂的内容特征直观地表达出来，使人一目了然。本文只针对篮球运动员的技术指标进行统计与分析，旨在实践计算机可视化技术在这一领域的应用，提高相关领域从业人员的工作效率[46-48]。

篮球运动员技术指标统计分析的特点：

（1）数据量较大：在我国注册的篮球运动员有数万人，而在篮球运动比我国发达的美国 NBA，以及其下的 ABA，CBA，NCAA 等联赛的注册运动员人数远远大于我国。

（2）分析内容具有多维性：除运动员的个人信息之外，对于其技术指标又有很多项，如进攻技术（可以再细分为：投篮技术、远投技术、罚篮技术、灌篮技术以及进攻意识等）、防守技术（可进一步细分为：篮板球技术、抢断技术、盖帽技术和防守意识等）和体能情况（身高、体重、弹跳能力、速度速度、反应能力、耐力以及力量等）。

（3）动态性：由于运动员的技术指标随训练水平、年龄增长、体能和伤病情况也在随时发生变化，因此运动员的技术指标存在随时间动态变化的特性。

为了较好的满足篮球运动员技术指标分析的需求，在设计时应满足以下客观要求：

（1）能够多角度全面而详细地显示数据：根据不同的需要按一定的程式对数据进行有效的统计分析，对于同一组数据能够从多角度进行显示；能够有效的进行数据的对比分析；根据需要能够以文本进行详细显示；

（2）快速、准确得到结果：应具有快速、准确、高效、更多更强表现力等特点，显示效果符号特征明显，易于识别，使用户对统计分析结果一目了然；

（3）良好的交互性、易于使用：使用者无须具备复杂的统计分析知识，就能够应用其进行篮球运动员技术指标分析并得到有效的结果；操作界面应简单易懂。

5.2 系统概述

本系统的主要目的是将用户获得的数据通过绘制图表来实现数据的可视性，主要需要解决两个问题：

（1）数据的动态统计。

（2）图表的生成。

统计分析的内容是多方面的，不同的用户观察的内容也是不同的，根据用户选择的条件，首先要对数据进行查询统计得到相应的统计结果，得到统计数据之后，按照用户选定的图表模式生成相应的图表。本系统使用 C++ Builder 作为开发工具，通过 BDE Administrator 接口来访问数据库，利用 C++Builder 的 Decision Cube 实现数据的统计功能和 C++ Builder 的 Steema TeeChart 7.07 组件生成图表。

5.2.1 Borland C++Builder 简介

C++ Builder 由著名的 Borland 公司开发，是 Windows 环境下功能最强大的 C++开发环境，它全面实现了 ANSI C++标准，提供了自己的扩展，并且兼容 PC 计算机上的两种最常用的 C++编译器，即 Borland C++和 Visual C++。Borland C++和 Visual C++的程序几乎不用作任何修改，就可以在 C++ Builder 下编译通过。C++ Builder 最显著的特点是它实现了 C++语言完全可视化开发，将 C++的面向对象和可视化紧密地结合起来，提供了一个功能强大、开发效率高的集成开发环境[49]。本系统之所以选择 C++ Builder 作为开发工具，主要基于其以下几个特点：

（1）真正可视化的 C++开发环境；

（2）高效存取数据库；

（3）ADOExpress 组件存取各类异质数据；

（4）丰富的图形设计；

（5）强大的调试功能。

5.2.2 Decision Cube 组件

在 Borland 公司的 C++Builder 中包含数据仓库决策立方体(Decision Cube)组件组，在这些组件的帮助下，用户可以很方便的建立数据仓库系统并实现 OLAP 操作。决策立

方体组件组主要包括：决策立方体(DecisionCube)组件、决策查询(DecisionQuery)组件、决策源(DecisionSource)组件、决策中枢(DecisionPivot)组件、决策栅格(DecisionGrid)组件和决策图表(DecisionGraph)组件组成，如图 5.1 所示。

Decision Cube 组件组可以与数据库基本表进行连接，反映数据的变化情况，对数据表进行数据统计、分析和图形显示。Decision Cube 组件组的应用十分灵活，而且功能十分强大，是一个良好的数据仓库应用系统开发工具。利用这些工具所开发出来的应用系统，对于管理决策具有良好的支持。而且可将数据仓库应用系统与管理信息系统的开发应用整合在一起，使数据仓库的应用与其他业务系统构成一个整体，从而提高了系统的开发效率，具体使用情况可以参见文献[48]。

图 5.1 Decision Cube 组件组

Fig. 5.1 Decision Cube components

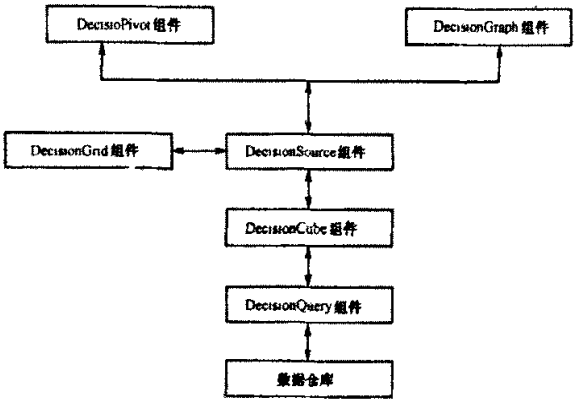


图 5.2 Borland 数据仓库各种组件关系图

Fig. 5.2 The relationship of all of Borland Data warehouse component

使用 Borland C++Builder 进行数据仓库系统开发，需要依靠 Decision Cube 组件组中的各种组件的相互配合，参见图 5.2。其中的 DecisionQuery 组件处于与物理数据库进行交互的地层。DecisionCube 组件主要对 DecisionQuery 组件从物理数据库中所取得的数

据进行分析，并将其转变为一个多维表的结构，然后通过 DecisionSource 组件交给 DecisionGrid 等组件进行显示。DecisionSource 组件在数据仓库的应用中起到了一个桥梁作用，将 DecisionCube 组件处理后的数据提交给 DecisionPivot，DecisionGrid，DecisionGraph 组件。DecisionPivot 组件主要用于对数据仓库中的数据操作进行导航，它提供一个便捷的按钮，便于用户对数据进行操作。DecisionGrid 组件主要用于对数据的分析结果进行显示，还可以改变数据显示区的颜色，以及数据的行、排列方式，即实现多维数据集的旋转分析。DecisionGraph 组件用来将所分析的数据以可视化的方式进行显示，有利于用户对数据进行直观的分析。

5.2.3 TeeChart 控件

TeeChart 控件是西班牙 Steema Software SL 公司的产品，它是由 David Berneda 用 Delphi 编写的，TeeChart 控件是集成在 C++ Builder 中的。它主要由 TChart，TDBChart，TQRChart 和 TDecisionGraph 这四个组件构成。其中 TChart 是最基本的核心组件，其他三个组件都是根据具体需要从 TChart 继承而来。TDBChart 在 TChart 的基础上添加了对数据库的支持。而 TQRChart 用于在 QuickReport 上绘制图表而开发的。TDecisionGraph 用于以图表的形式支持数据决策。使用 TeeChart，我们可以绘制出下列图样：

(1)点状图像：点状图 Point，彩点图 DeltaPoint，三维点图 Point3D,形象点图 ImagePoint；

(2)线状图像：折线图 Line，多条折线图 Lines，8 条折线图 Lines8,箭头图 Arrows，贝塞尔曲线图 Bezier，等高线图 Contour，线点图 LinePoint；

(3)面状图像：三角表面图 TriangleSurf，金字塔图 Pyramid,瀑布图 WaterFall，高低图 HighLow，区域图 Area，形状图 Shape；

(4)柱状图像：直方图 Bar，三维柱图 Bar3D，连柱图 BarJoin，形象柱图 ImageBar，容积图 Volume，蜡烛图 Candle，误差柱图 ErrorBar，误差图 Errors，漏斗图 Funnel，柱图 Histogram；

(5)梁状图像：进度图 Gantt，横条图 HorizBar；

(6)圆形图像：饼图 Pie，圆环图 Donut，气泡图 Bubble，时钟图 Clock；

(7)方形图像：箱点图 BoxPlot，箱点横图 BoxPlotH，日历图 Calenda，彩格图 ColorGrid；

(8)网状图像：雷达图 Radar，极地图 Polar，风向频率图 WindRose，史密斯图 Smith。

TeeChart 类的属性和方法分析。TeeChart 的主类是 TChart, TChart 中使用了 56 个类、325 个属性、125 个方法以及 28 个事件，这使得 TChart 具有非常强大的功能。下面简单介绍其中一些比较重要类的属性和方法：

(1) TChart. Series：序列数组类，是要显示数据的主体。在一个图表中可以有一个或多个序列，每个序列可以有不同的显示类型，如 Line, Bar, Pie, Arrow 等等。

(2) TChart. Axes：坐标轴类，控制图表坐标轴的属性，在缺省的情况下，坐标轴可以自动地根据不同的数据设置好标度范围和间隔，当然也可以手工调整。

(3) TChart. Legend：图例类，控制图表的图例显示。Legend 是图表中的一个长方形的用来显示图例标注的区域。可以标注 Series 的名称或者 Series 中的项目和数值。

(4) TChart.Panel：面板类，可以设置图表的背景。可以使用渐变的颜色或者图像文件作为整个图表的背景；

(5) TChart. Canvas：画布类，Canvas 可以让设计者绘制自己的图形。使用方法和 Delphi 中的 Canvas 样。有 Arc, LineTo, Polyline, TextOut 等各种画图的方法可以调用。

TChart 的一些属性实际上是其他类的变量，这些类又具有自己的属性和方法。如 Ititles 类又具有 Text, Color, Font 等属性，可以用这些属性来设置题头的文本、颜色和字体。

TeeChart Pro 是一个更为专业的 VCL 图表控件，支持几百种二维和三维图表风格，并提供 40 多个数学和统计函数、无限制的轴和 22 个调色板。TeeChart 还集成了打印预览，图表可导出为 JPEG, EPS, PDF, PNG, PCX, GIF, Bitmap 和 metafile 文件。同时还可提供.NET, ActiveX 和 COM 版本。TeeChart 还包括一个强大的、完整的编辑对话框，几乎可用于每个组件和子组件，允许快速的设计复杂图表应用程序。TeeChart Pro 为缩减可执行程序大小被分成完全面向对象的多个模块。它还允许开发者创建自定义包组成他们自己需要的模块。

5.2.4 Chart Fx 组件

Chart FX 是 Software FX 公司的产品。Software FX 公司成立于 1993 年，开发者提供适用于.NET, COM, IT|SQL 和 Java 的提供高质量，性能稳定和高可用性的关键数据可视化工具。目前，Software FX 为不同开发环境开辟了适应于各种开发环境多条产品线，其产品包括：Chart FX for Visual Studio 2005、Chart FX for .NET 6.2、Chart FX Internet 6.2、Chart FX for Java 6.2 、Chart FX Developer Studio 2006 等。

Software FX 同样为 Borland C++Builder 系列产品提供组件，与 TeeChart 同样具有强大的功能，用户可以通过属性编辑窗口控制 Chart FX 组件的显示，如图 5.3 所示。

图 5.3 Chart FX 属性编辑对话框

Fig. 5.3 Chart FX Attribute Dialog Box

Chart FX 组件将各种相关属性进行详细分类，各类属性被安排在不同的选项卡中，用户可以对组件进行动态的控制。

(1) Appearance 选项卡：允许用户改变显示图样的重要特征。通过编辑组件的 ChartType 属性可以控制图中的显示图样：Line（线型图）、Bar（条状图）、Spline（曲线图）、Pie（饼图）、Scatter（散点图）等。

(2) 3Dview 选项卡：控制用户的定制的图形的三维显示效果，还可以帮助用户动态的指定观察坐标轴的视点。

(3) Data Values 选项卡：允许用户改变或指定图中所包含属性的数值，按照用户的要求对数据进行编辑。

(4) Tools 选项卡：帮助用户显示、隐藏以及定制一些管理可视化图样的工具，其中包括：打印、保存、图样控制等。

(5) Lines 选项卡：允许用户改变图中的线条的线型，包括：线的类型、颜色以及宽度等。

(6) Style 选项卡：用来指定一些图的外观以及文件和剪贴板输出方面的属性。

(7) Label 选项卡：用来填写图中的一些说明性的文字。

(8) Element 选项卡：帮助用户定制一些辅助观察的元素。

(9) Color 选项卡：用于设定图中各种元素的颜色。

(10)Font 选项卡：用于设定图中各种文字的字体。

5.3 系统设计

本篮球运动员技术指标统计分析系统，具有数据管理、数据统计分析、数据可视化表示以及可视化交互控制等功能模块，参见图 5.4。数据管理模块负责常规的数据增删改，以及数据挖掘前数据的预处理功能，并对数据进行及时的维护。统计分析模块主要负责对数据进行汇总处理，实时的对数据库中的数据进行全方位的统计处理，并以图表或图形的方式为用户进行显示。可视化模块以图标可视化、几何变换可视化等多种技术实现多维数据的可视化显示，为用户观测数据提供多种服务。维度控制模块实现上文提到的维数控制与维排列顺序控制算法，对可视化过程和结果进行控制。

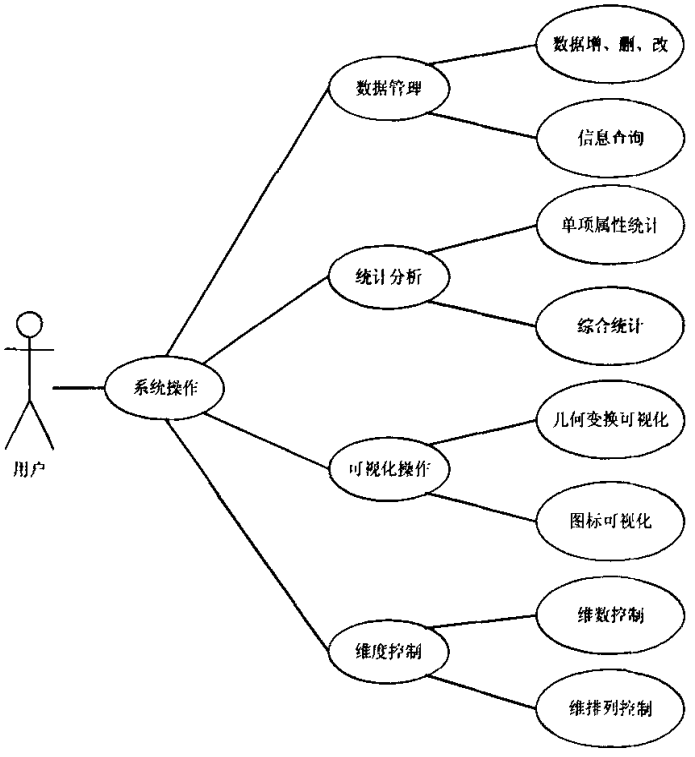


图 5.4 系统用例图

Fig. 5.4 Use Case of System

在完成对系统功能的分解之后需要对系统的工作流程进行设计，系统的工作流程参见图 5.5 系统活动图，具体做法参见文献[50,51]。图 5.5 包含了系统运行过程中的一些较为典型的活动，通过这些活动用户可以便捷的对数据进行汇总、筛选以及观察分析，最终实现获取信息的目标。

（1）据源选择：用户给定数据库别名，选择数据库路径、数据库类型和表名称后，由系统完成别名的创建。

（2）数据源的更新：每隔一段时间间隔，系统对用户所指定的数据源进行更新，并对相关的统计值进行汇总。

（3）统计分析：通过对统计分析条件设置来实现对数据来实现对特定主题的分析，其设置工作主要包括两部分：

1）筛选条件选择：筛选条件作为从源数据中筛选出用户需要的数据记录的条件，在这里还包括对多个条件之间组合方式的选择。

2）统计内容选择：指定要对哪部分内容进行可视化处理，筛选出相应字段。

（4）数据可视化：以表格的形式对数据进行显示，并进行数据汇总，使用户可以较为直观的对数据相应属性的取值进行研究。

（5）图表类型选择：将统计分析或数据可视化的结果进行汇总，并按照用户的意愿进行定制。用户可以选择用于统计分析的柱状图、条状图以及饼图等对统计分析的结果进行显示，也可以利用平行坐标图、雷达图以及散点图等对多维数据进行可视化显示。

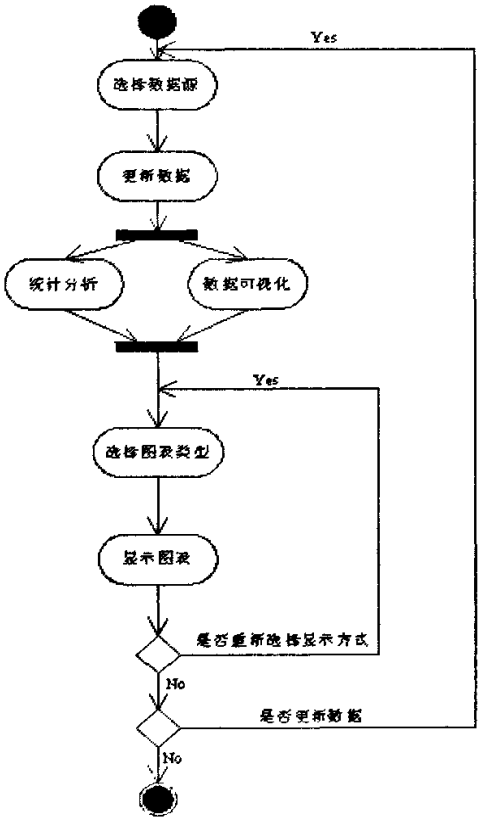


图 5.5 系统活动图

Fig. 5.5 Active diagram of System

（6）图表可视化：获取用户对图表的定制信息，对多维数据或者数据的汇总值进行显示。利用不同的可视化技术对多维数据进行多方面内容的对比，用户可以快捷地了解数据内部隐含的信息。

5.4 统计分析和可视化模块的实现

本系统的数据管理模块与传统的信息管理系统区别不大，该模块主要功能是对数数据进行管理，并同时完成数据挖掘和可视化数据的预处理工作。这里主要介绍统计分析功能模块以及可视化模块的实现方式。

5.4.1 统计分析的实现

一般对数据库的查询是通过记录过滤和查询语句来实现的，通常数据库查询的三个步骤为：

（1）筛选记录；

（2）形成子表；

（3）子表内完成数据处理。

在这里由于统计分析的内容是多方面的，如果单独使用 SQL 语句来实现查询将较难适应复杂的情况，也不能完全满足所有情况下的需要。复杂的原因如下：

（1）统计分析的内容不确定；

（2）筛选的条件中可以包含多个子项，比如用户希望在记录集中寻找包含第一、二、三项指标达到用户期望，三项指标均达标，根据用户的选择还可包含“总体”选项；

（3）可以只选择某个条件中的一个子项，比如只选择达标程度中第一项达到用户期望，再同时满足其它条件下对某些信息进行统计。

在上述三种情况下，数据记录的查询较为复杂，可以采用前文提到的 Decision Cube 组件来完成复杂的记录筛选过滤等方面的高级需求。

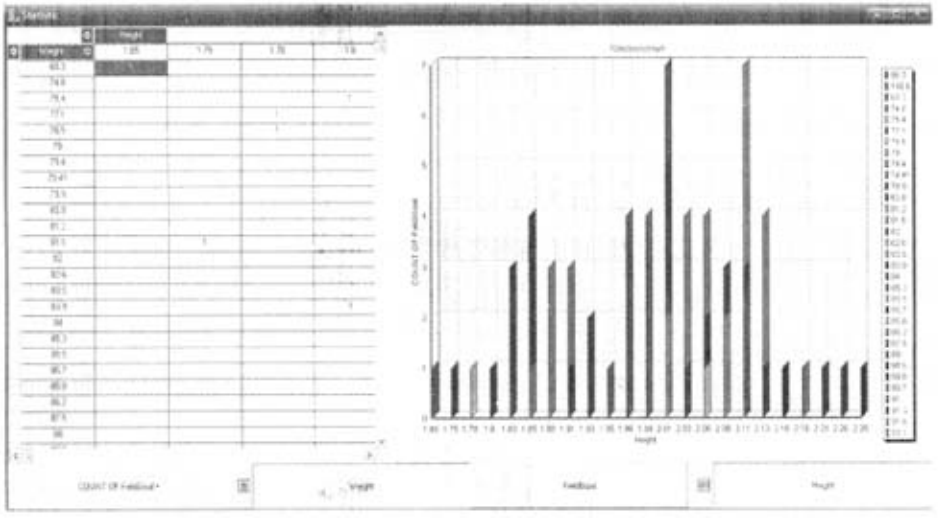


图 5.6 采用 Decision Cube 组件实现数据统计分析功能

Fig. 5.6 Statistical analysis using Decision Cube components

采用 C++Builder 中的 Decision Cube 组件可以方便的对多维数据进行多个方面进行统计，用户可以从多个侧面分析数据对数据进行汇总，为最终的决策提供有力支持。按照上文提到的组织结构安排组件的层次并完成相关的属性设置便可以完成复杂的数据筛选和查询，如图 5.6 所示。用户可以控制 DecisionPivot 按钮来对于目标数据维度的汇总信息和可视化显示进行控制。用户还可以与 DecisionGrid 进行交互，获取相关维度上数据的汇总信息。

5.4.2 多维数据的可视化显示

（1）平行坐标图：通过平坐标图可以将运动员的技术指标显示在一个二维空间内，更加便于用户对数据进行分析与理解。在图 5.7 中可看到以全部数据为显示对象而生成的图表。从表面看来图表中的图线较为杂乱，不利于用户进行观察。通过控制篮球运动员的分类按钮可以对显示对象进行控制，从而更有利于用户的观察分析。图 5.8 是对数据进行过滤以后得到的可视化效果。可以看出经过筛选后的数据量明显减少，并且规律性也比较强。

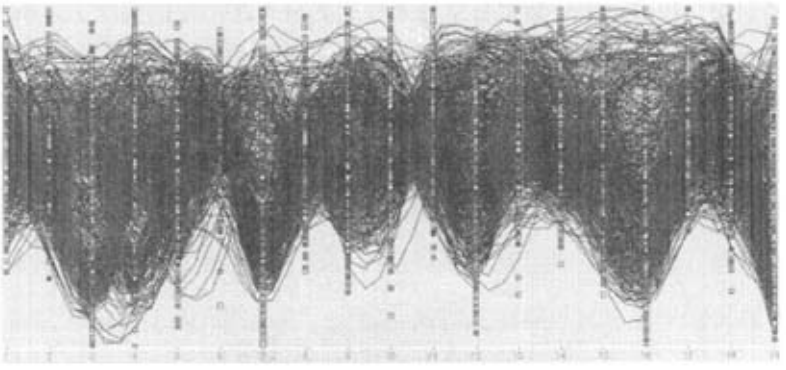


图 5.7 多维数据在平行坐标中的显示

Fig. 5.7 Diagram of Multidimensional data in Parallel coordination

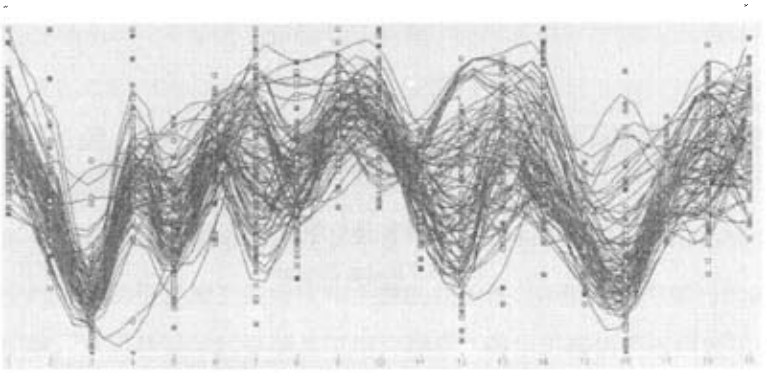


图 5.8 经过筛选处理的平行坐标图

Fig.5.8 Diagram of Multidimensional data in Parallel coordination after

（2）雷达图：又称为星图或蜘蛛网图，可以在二维平面上表示高维数据，是目前应用最广泛的对多元资料进行作图的方法，利用雷达图可以直观的反映各样本点之间的关系并进而对样品进行归类参见图 5.9。图 5.9 中是三名篮球运动员的 19 个技术指标的雷达图，从图中可以很明确的发现每名运动员之间技术指标差异。由于雷达图的范围有限，因此利用雷达图同时显示大量数据的效果是不理想的。于是我们采用雷达图来进行篮球运动员之间技术指标的比较。利用可视化的显示，用户可以直观的得到相关的比较结果。

在利用平行坐标图、雷达图以及柱状图显示数据时，由于图线较多有时可能给用户的观察带来不便，可以利用对比明显的颜色标识不同数据记录所对应的曲线，使用户更

区分不同的记录。也可以采用数据分层技术使数据分布在不同的层次上，系统可以按照用户的需求在不同的数据层次上完成数据的上卷下钻操作。

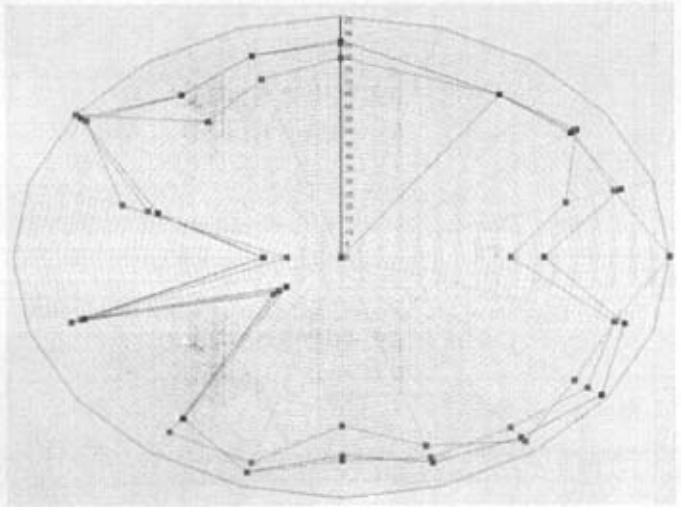


图 5.9 雷达图

Fig. 5.9 Radar diagram

通过本系统可以使用户方便的对运动员的技术信息进行观察和分析，从而得到可靠的观察结果指导对运动员的选拔和培养进行指导。本系统具有合理的扩展接口可以进一步扩充数据挖掘算法以及知识库等功能模块，使系统具有更强的推理和学习功能，能对数据进行更加深入的研究得出更多更全面的挖掘结果。

6 结论

信息技术和网络技术迅速发展使数据的规模日益巨大。如何有效处理这些数据，将数据转换为可理解的信息，成为一个严峻的问题。在数据挖掘过程中充分利用可视化技术对于提升数据可用性和预测准确度有着十分重要的意义。通过对本课题的研究，对数据可视化及其实现技术有了充分的了解，同时对计算机图形图像处理、数理统计学、数据挖掘等方面的知识有了较深的认识。分析了数据挖掘和可视化技术的发展现状，对数据挖掘过程中的数据模型以及流行的可视化技术进行深入研究。

通过对可视化过程中的交互技术现状的研究与分析，本文提出在交互过程中，针对多维数据的维在显示过程中缺乏定量指导的问题，提出利用多维数据维相似程度来指导数据维度在可视化过程中的排列，使相似的维度排列在比较接近的位置，使可视化结果更接近用户所关心的主题，最终使用户可以合理的观察数据并对数据的规律进行高效总结。

对于多维数据在显示过程中维数众多不易于用户观察的问题，采用相关分析算法对分析各个维度对主题的信息增益或维度与主题的相关性，并根据用户提出的阈值对维数的显示进行控制，从而使显示结果更易于用户观察，得出更准确的观察结果。

本课题还实现了一个篮球运动员技术指标分析系统，其中实现了多种实用的可视化技术，如几何转换可视化和图标可视化技术等，并利用维相似度算法和属性相关分析对数据在显示中的维度进行控制，系统可以对大量数据进行动态统计，然后绘制各种图表，支持多种交互手段，以便用户直观、快速对数据进行理解与分析。

参 考 文 献

[1] Jiawei Han, Micheline Kamber. 数据挖掘概念与技术. 北京：机械工业出版社，2001.

[2] B. H. McCormic, T. A. DeFanti, M. D. Brown, et al. Visualization in Scientific Computing. Computer Graphics, 1987, 21(6):45-50.

[3] 孙家广. 计算机图形学. 北京：清华大学出版社, 1998.

[4] 章毓晋. 图像处理和分析基础. 北京：高等教育出版社, 2002.

[5] 石绥祥, 夏登文, 于戈. DSS模型系统建模与实现方法研究及其在海洋中的应用. 海洋通报，2005, 24(1):69-75.

[6] 戴秀丽，孙成. 太湖沉积物中重金属污染状况及分布特征探讨. 上海环境科学，2001, 20(2):71-74.

[7] 王衍. 基于信息可视化技术的税务决策支持系统分析. 数量经济技术经济研究，2004, (4):148-153.

[8] Susan E. George. A Visualization and Design Tool (AVID) For Data Mining With The Self-Organizing Feature Map. International Journal on Artificial Intelligence Tools, 2000, 9(3):369-375.

[9] C. Stolte, D. Tang, P. Hanrahan. Polaris: A System for Query, Analysis and Visualization of Multidimensional Relational Databases. IEEE Trans. Visualization and Computer Graphics, 2002, 8(1):52-65.

[10] Daniel A. Keim, Ming C. Hao, Umeshwar Dayal. Hierachical Pixel Bar Chart. IEEE Transaction and Computer Graphic, 2002, 8(3):255-269.

[11] Eick, S. G. Steffen, J. L. Sumner, et al. Seesoft-A Tool for Visualizing Line Oriented Software Statistics. IEEE Transactions on Software Engineering, 1992, 18(11): 957-968.

[12] D. A. Keim, H.-P. Kriegel. Visdb: Database Exploration Using Multidimensional Visualization. Computer Graphics & Applications, 1994, (6):40-49.

[13] 傅德胜，傅涛. 可视化数据挖掘技术. 教育信息化, 2005, (8):77-78.

[14] 黄江涛，刘自伟，黄晓芳. 用于数据挖掘的多维数据可视化技术. 兵工自动化, 2005, 24(3):52-53.

[15] Michael D. Lee, Marcus A. Butavicius, Rachel E. Reilly. Visualizations of binary data: A comparative evaluation. Int. J. Human-Computer Studies, 2003, 59:569-602.

[16] Alfred Inselberg. Visualization and data mining of high-dimensional data. Chemometrics and Intelligent Laboratory Systems, 2002, 60:147-159.

[17] Trevor D. Collins. Applying software visualization technology to support the use of evolutionary algorithms. Journal of Visual Languages and Computing, 2003, 14:123-150.

[18] George W. Furnas, Andreas Buja. Prosection Views: Dimensional Inference through Sections and Projections. Journal of Computational and Graphical Statistics, 1994, 3(4): 363-367.

[19] 刘天桢, 童恒庆. 基于投影寻踪和聚类分析的多维数据可视化. 福建电脑, 2005, (8):113-114.

[20] Witold Dzwinel, Jan Błasiak. Method of particles in visual clustering of multi-dimensional and large data sets. Future Generation Computer Systems, 1999, 15:365-379.

[21] Ioannis Kopanakis, Babis Theodoulidis. Visual data mining modeling techniques for the visualization of mining outcomes. Journal of Visual Languages and Computing, 2003, 14: 543-589.

[22] 周晓峥, 刘勘, 孟波. 多维数据集的平行坐标表示及聚簇分析. 计算机工程, 2002, 28(1):94-143.

[23] 王绍敏, 孙晓静, 王克峰, 等. 应用平行坐标系进行可视化优化设计. 计算机与应用化学, 2004, 24(1):11-15.

[24] Daniel A. Keim. Information Visualization and Visual Data Mining. IEEE Transaction and Computer Graphic, 2002, 8(1):1-8.

[25] Maria Cristina Ferreira de Oliveira, Haim Levkowitz. From Visual Data Exploration to Visual Data Mining: A Survey. IEEE Transaction and Computer Graphic, 2003, 9(3):378-394.

[26] Y-H. Fua, M. O. Ward, E. A. Rundensteiner. Hierarchical Parallel Coordinates for Exploration of Large Datasets. In Proc. of IEEE Visualization, 1999:43-50.

[27] Daniel A.Keim. Designing Pixel-Oriented Visualization Techniques: Theory and Applications. IEEE Transaction on Visualization and Computer Graphics, 2000, 6(1):59-78.

[28] 任永功, 于戈. 一种支持可视化数据挖掘的图形后处理方法. 小型微型计算机系统, 2005, 26(11):1955-1959.

[29] Jing Yang, Matthew O. Ward, Elke A. Rundensteiner. Interactive hierarchical displays: a general framework for visualization and exploration of large multivariate data sets. Computers & Graphics, 2003, 27:265-283.

[30] Tamara Munzner. Information Visualization. IEEE Computer Graphics and Applications, 2002, 22(1):20-21.

[31] M.O. Ward. Xmdvtool: Integrating Multiple Methods for Visualizing Multivariate Data. Proc. Visualization '94, 1994:326-336.

[32] 水俊峰, 陈树晓, 郭茂林, 等. 基于数据仓库的政府决策支持系统研究. 科技情报开发与经济, 2005, 16(14):210-212.

[33] Daniel A.Keim. Visual Exploration of Large Telecommunication Data Sets. IEEE Transaction on Visualization and Computer Graphics, 1999, 19(3):16-19.

[34] 任永功, 于戈. 一种多维数据的聚类算法及其可视化研究. 计算机学报, 2005, 28(11):1861-1865.

[35] Chaouki Daassi, Laurence Nigay, Marie-Christine Fauvet. Visualization Process of Temporal-Data. Database and Expert Systems Applications, 15th International-Springer, 2004: 914-924.

[36] Steven Morris, Camille DeYong, Zheng Wu, et al. DIVA: a visualization system for exploring document databases for technology forecasting. Computers & Industrial Engineering1, 2002, 43:841-862.

[37] Emden R. Gansner, Stephen C.North. An open graph visualization system and its applications to software engineering. SOFTWARE—PRACTICE AND EXPERIENCE Soft. Pract. Exper., 2000, 30(11):1203-1233.

[38] Gennady Andrienko, Natalia Andrienko. Knowledge-Based Visualization to Support Spatial Data Mining. Advances in Intelligent Data Analysis: Third International Symposium, IDA-99, 1999, 1642:149-160.

[39] Tubao Ho, Trongdung Nguyen, Ducdung Nguyen, et al. Visualization Support For User-Centered Model Selection in Knowledge Discovery and Data Mining. International Journal on Artificial Intelligence Tools, 2000, 1(10):691-713.

[40] 赵欣，赵海，徐凌宇. 基于Data Warehouse 技术的数据可视化的应用. 东北大学学报，2000, 21(4):365-367.

[41] 孙泳，刘少辉，史忠植. 数据仓库中多维分析的数据展现. 计算机工程与应用, 2004, (4):174-177.

[42] 华丽, 肖美添. 样本筛选与操作优化的可视化方法实现. 华侨大学学报, 2005, 26(1):76-79.

[43] 张德锋, 郭玉霞, 宋志刚. 数据挖掘技术. 航空计算技术, 2005, 35(3):76-79.

[44] 夏登文，石绥祥，于戈，等. 海洋数据仓库及数据挖掘技术方法研究. 海洋通报，2005, 24(3):60-65.

[45] 石昊苏, 韩丽娜. 数据可视化技术及其应用展望. 2005年全国自动化新技术学术交流会会议论文集, 2005:180-183。

[46] 阎守扶, 汪安, 刘东艳, 等. 关于体能类速度力量项群和速度耐力项群女运动员头发中微量元素的多变量样本图分析法. 北京体育大学学报, 1999, 22(3):46-49.

[47] 刘文新, 奕兆冲, 汤鸿霄. 应用多变量脸谱图进行河流与湖泊表层沉积物重金属污染状况的综合对比研究. 环境化学1997, 16(1):23-29.

[48] 陈京民. 数据仓库与数据挖掘技术. 北京:电子工业出版社, 2002.

[49] 席卫文, 张春晓, 李光明. C++Builder6程序设计与实例. 北京:冶金工业出版社, 2003.

[50] Roger S. Pressman. Software Engineering Apractitioner' s Approach(Fifth Edition). 北京:机械工业出版社, 2002.

[51] Grady Booch, James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide. 北京:机械工业出版社, 2001.

在学研究成果

张文鹤. 一种多维数据可视化模型技术的研究. 沈阳工业大学学报增刊. [已录用]

致　　谢

本文是在牛连强教授的精心指导下完成的。牛老师孜孜以求的治学精神，细致灵活的科研思想，和蔼可亲的为人作风，不仅激励着我完成课题研究工作，而且使我从中学到了更多宝贵的东西，必将使我终身受益。

从课题开题至研究工作的完成，得到了张胜男副教授的细心指导，张老师诲人不倦的精神使我深受感动，在以后的日子里也将永远激励着我。感谢给予我帮助和鼓励的同学们。

最后向在百忙之中抽出宝贵时间评审本论文的专家、学者致以最诚挚的谢意！