学号 142006010324

年级 2014级



**本科毕业论文**

**基于多源数据的高速公路短时交通流量预测**

|  |  |
| --- | --- |
| **专 业** 计算机科学与技术 |  |
| **姓 名** 邵程立 |  |
| **指导教师** 谢在鹏 博士 |  |
| **评 阅 人** |  |

**2018年5月**

**中国 南京**

**BACHELOR'S DEGREE THESIS**

**OF HOHAI UNIVERSITY**

**Short-term Traffic Flow Forecast Based on Multi-source Data**

College : College of Computer and Information

Subject : Computer Science and Technology

Name : ChengLi Shao

Directed by : ZaiPeng Xie Lecturer

NANJING CHINA

**郑 重 声 明**

本人呈交的毕业论文，是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料真实可靠。尽我所知，除文中已经注明引用的内容外，本设计（论文）的研究成果不包含他人享有著作权的内容。对本设计（论文）所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确的方式标明。本设计（论文）的知识产权归属于培养单位。

本人签名： 日期：

摘 要

交通流量预测是测量交通状态的一种方式， 实时的交通状态对交通控制和智能交通有重要意义。交通流量预测一般分为三种：短期、中期、长期。由于交通流量具有波动性，出于预测的精确度，研究短时交通流量更有意义。

短时交通流量预测可以看着是一种非线性回归问题，从历史数据中挖掘出有迹可循的规律或者说成是学习拟合函数的各种参数，并将其应用于预测未来数据。大量长期有效的数据是实现这一目标的基础。传统的交通流量预测主要是根据历史的交通流量进行分析，而忽视交通流量在一定程度上的相关性，导致其输入单一。为了提高交通流量预测的准确性，用多源数据进行短时交通流量预测，增加各种影响因素，从而提高准确率。多源数据，例如，环境因素（降雨量，温度，光照），收费站进出口数。

由于交通流量的非线性，同时神经网络在处理非线性问题上有着先天的优势。【Hornik et al.,1989】证明，只需一个包含足够多神经元的隐层，BP网络就能以任意精度逼近任意复杂度的连续函数。此外SVR，集成学习等方法在非线性处理上也有一定效果。因此在算法的选择上，结合多种优质算法进行预测。研究对象的选择上，由于高速公路的路网结果相对其他道路比较简单，相关制约因子容易理解。故选择高速公路作为实践对象。

因此，本文将使用多源数据进行短期流量预测，实现了基于BP神经网络的多种回归算法相结合的模型构成的短期流量预测系统。并制作了一个简单的UI进行相关操作。

短期流量预测，对智能交通系统的发展有重要意义，而其预测的精确度更是衡量智能交通系统的性能。

**关键词** 交通状态；短时交通流量；多源数据；SVR；集成学习；神经网络

**ABSTRACT**

Traffic flow forecasting is a way of measuring traffic conditions. Real-time traffic conditions are important for traffic control and intelligent traffic. Traffic flow prediction is generally divided into three types: short-term, medium-term, and long-term. Due to the volatility of traffic flow, it is more meaningful to study short-term traffic flow for the accuracy of the prediction.

Short-term traffic flow forecasting can be viewed as a non-linear regression problem. Historical rules can be traced out or learned as various parameters for learning a fitting function, and used to predict future data. A large amount of long-term effective data is the basis for achieving this goal. The traditional traffic flow forecasting is mainly based on the analysis of historical traffic flow, while ignoring traffic flow to a certain degree of correlation, resulting in a single input. In order to improve the accuracy of traffic flow forecasting, short-term traffic flow forecasting using multi-source data, Improves accuracy by various influencing factors. Multi-source data, for example, environmental factors (rainfall, temperature, light), import and export number of toll stations.

Due to the nonlinearity of traffic flow, neural networks have inherent advantages in dealing with non-linear problems. [Hornik et al., 1989] demonstrated that a BP network can approximate a continuous function of arbitrary complexity with arbitrary precision with only one hidden layer containing enough neurons. In addition, methods such as SVR and ensemble learning also have certain effects on nonlinear processing. Therefore, combining a variety of high-quality algorithms for prediction, the choice of research objects, because the road network results of the highway is relatively simple compared to other roads, and the relevant constraints are easy to understand. Therefore, the highway is chosen as a practical object.

Therefore, this paper will use multi-source data for short-term traffic forecasting and implement a short-term traffic forecasting system based on a combination of multiple regression algorithms based on BP neural networks.

Short-term traffic forecasting is of great significance to the development of intelligent transportation systems, and the accuracy of its predictions is a measure of the performance of intelligent transportation systems (ITS).

**Keywords:** Traffic Status; Short-term Traffic Flow; Multi-source Data; SVR ; integrated learning ; NN

**目录**

[摘 要 I](#_Toc452843791)

[ABSTRACT II](#_Toc452843792)

[目录 III](#_Toc452843793)

[第1章 绪论 1](#_Toc452843794)

[1.1研究背景 1](#_Toc452843795)

[1.2国内外研究现状 1](#_Toc452843796)

[1.3本文工作 3](#_Toc452843797)

[第2章 基于DEM的数字流域特征及提取 4](#_Toc452843798)

[2.1 DEM简介 4](#_Toc452843799)

[2.2 数字流域特征指标 4](#_Toc452843800)

[2.2.1 数字流域特征相关概念 4](#_Toc452843801)

[2.2.2 基于DEM提取的数字流域特征 6](#_Toc452843802)

[2.3 数字流域特征提取 8](#_Toc452843803)

[2.3.1 数据准备 8](#_Toc452843804)

[2.3.2 工具介绍 8](#_Toc452843805)

[2.3.3 提取步骤 8](#_Toc452843806)

[2.4 小结 17](#_Toc452843807)

[第3章 流域相似性分析的相关算法 19](#_Toc452843808)

[3.1 主成份分析 19](#_Toc452843809)

[3.2 K-Means算法 20](#_Toc452843810)

[3.3 层次聚类算法 20](#_Toc452843811)

[3.4 相似流域分析实例 21](#_Toc452843812)

[3.4.1 数据准备 21](#_Toc452843813)

[3.4.2 数据分析 22](#_Toc452843814)

[3.5 小结 27](#_Toc452843815)

[第4章 基于DEM的相似流域查找系统的设计 28](#_Toc452843816)

[4.1 需求分析 28](#_Toc452843817)

[4.2 总体设计 28](#_Toc452843818)

[4.2.1 系统结构设计 28](#_Toc452843819)

[4.4.2 系统流程设计 30](#_Toc452843820)

[4.3 系统功能设计 30](#_Toc452843821)

[4.4数据库设计 31](#_Toc452843822)

[4.4.1河流相似性分析基本信息表 31](#_Toc452843823)

[4.4.2河流相似性性分析属性字典表 33](#_Toc452843824)

[4.5详细设计 33](#_Toc452843825)

[4.5.1 单流域搜索 33](#_Toc452843826)

[4.5.2 多流域分析 34](#_Toc452843827)

[4.5.3 DEM参数文件上传 35](#_Toc452843828)

[4.5.4 查询河流信息 36](#_Toc452843829)

[4.5.5 更新河流信息 36](#_Toc452843830)

[4.5.6 新增河流信息 36](#_Toc452843831)

[4.5.7 界面设计 37](#_Toc452843832)

[4.6 小结 38](#_Toc452843833)

[第5章 基于DEM的相似流域查找系统的实现 39](#_Toc452843834)

[5.1 系统开发环境 39](#_Toc452843835)

[5.2 系统实现 39](#_Toc452843836)

[5.2.1 单流域搜索功能实现 39](#_Toc452843837)

[5.2.2 多流域分析功能实现 41](#_Toc452843838)

[5.2.3 上传DEM参数文件功能实现 42](#_Toc452843839)

[5.2.4 河流信息管理功能实现 46](#_Toc452843840)

[5.2.5 系统界面实现 47](#_Toc452843841)

[5.3 系统展示 48](#_Toc452843842)

[5.3.1 系统界面展示 48](#_Toc452843843)

[5.3.2 单流域搜索功能 49](#_Toc452843844)

[5.3.3 多流域分析功能 50](#_Toc452843845)

[5.3.4 河流信息查询功能 51](#_Toc452843846)

[5.3.5 上传DEM参数文件功能 52](#_Toc452843847)

[5.3.7 新增河流信息功能 52](#_Toc452843848)

[5.3.8 编辑河流信息功能 53](#_Toc452843849)

[5.4 小结 53](#_Toc452843850)

[第6章 总结与展望 54](#_Toc452843851)

[致 谢 55](#_Toc452843852)

[参考文献 56](#_Toc452843853)

[附录1：国外经典论文——原文 58](#_Toc452843854)

[附录2：国外经典论文——翻译 69](#_Toc452843855)

**第1章 绪论**

**1.1研究背景**

近年来，随着我国物质生活水平的提高，人们对于我国水文行业正从传统水文向现代化水文迈进，“十一五期间”，全国水利信息化实现了实时水雨情、防洪工程、基础水文数据库等专题数据库，初步在全国范围内形成了体系较为完整、内容相对丰富的水利信息存储体系。同时，基础地理信息资源开发利用也取得了一些成果，已完成第一代1：250000全国陆地部分满覆盖水利基础数字地图数据库，基本建成覆盖我国东部地区的1：50000水利基础数字地图数据库。数字流域综合应用遥感(RS)、地理信息系统(GIS)、全球定位系统(GPS)、虚拟现实(VR)、网络和超媒体等现代高新技术,实现了流域信息的釆集和数字化管理,构建了可视化的流域综合管理平台和流域三维仿真模型[1]。随着高分辨率遥感卫星、机载传感系统、全球卫星导航系统、合成孔径雷达等对地观测技术的发展,DEM数据每天都在以TB为单位级的数量增加，DEM空间数据步入大数据世代。在数字高程模型（DEM）、地理信息系统（GIS）和电子信息技术的支持下，以数字高程模型为基础，可以提取出自然界流域的形状、坡度、大小，河流流量、流向、汇流量等一系列数字流域特征，为水文分析提供基础。

中国幅员辽阔，尽管全国各地有36720处水文测站，但站网密度分布并不均匀。中小河流范围广大，但是大部分流域存在测站密度稀疏，甚至没有测站的情况，缺乏采集数据、监控水情的技术手段，是典型的少资料或无资料流域。从数据中挖掘出有价值的规律和模式及准确的水文预测等是现代水文工作的目标，而这些都需要大量的长期的数据资料做支撑[2]。2003年国际水文科学协会上提出的“无测站流域水文预测”（PUB）中提出了“移植法”，即如果有资料区的自然地理环境与研究区域相似，则直接移用该地区的资料到无资料区。水文相似性分析是解决无资料问题的一个重要方法。通过数据挖掘技术比较分析不同流域间的水文特征，对相似流域进行查找，或者多个流域进行分类比较，对水文资料的补充、中小流域防洪减灾、流域治理、水土保持等具有重要意义。

**1.2国内外研究现状**

流域相似性研究是水文科学的前沿问题，对流域综合性研究治理有着重要意义。研究的展开将加强人们对相似流域的理解，根据流域特点进一步对流域类型进行总结，提出研究模型，完善水文资料，提高流域研究的科学性。其对无资料或少资料地区洪水预测、流域治理、生态环境监测等有重要指导意义。

数字高程模型从上世纪60年代开始被广泛应用于测绘、防洪、水文模拟、土木工程、规划、地质等领域。基于数字高程模型数据，可以提取出流域面积、流域平均坡度、河网密度、流域圆度比、主河道长度、河道分支频率、流域相对高差、伸长比、流域长度和流域形状因子等下垫面信息指标。利用数字高程模型构建数字流域模型并提取流域水文特征，是研究流域相似性的重要步骤。1975年Peuker等提出基于谷点识别方法[3]，这是最早出现的提取流域特征的方法，它会生成很多不连续的河道片段；1984年，Mark等提出TOPAZ（Topographic PArame-teriZation）方法[4]，其中生成河网采用的是D8法来判断流向；O'Callaghan等提出坡面流模拟法[5]，依据水总是沿斜坡最陡方向流动的原理来实现流域的自动提取，多常用平滑DEM处理方法；Tribe提出基于格网DEM的栅格格网水流流向法[3]，利用矩形窗口扫描DEM矩阵，通过流向判断河网位置；Jenson和Domingue提出了D8算法[6]，它被广泛应用于流向分析和洼地填充；Quinn 等1991 年提出多流向法[7]。国内学者从90年代开始对基于DEM提取流域特征进行研究，赵美玲、张永红、张振[8-10]等利用ArcGIS软件，张国义等[11]运用RiverTools工具对DEM数据进行处理，自动提取流域特征指标及河网信息，对流域特征进行研究。

相似流域研究已经从最初的依据经验判断的传统方法发展到根据可计算的数学模型来进行流域相似度比较，使流域相似研究更加严谨和系统化。国外学者从上世纪50年代开始定量化的水文相似研究。Langhaar将物理模型和量刚分析引入水文研究中[12]；Valdes和Rodriguez-Iturbe[13]通过链接瞬时单位线（IUH）与流域地貌参数来建立地表径流的水文响应函数；Andrew R. Young开发改进了MOPEX先验模型的参数估计技术[14]，通过研究英国的260个大型数据集，用基于最邻近方法和基于流域特征的回归模型参数两个区域化方法进行评估，最终提出了一种基于水文响应单元替代建模框架。国内学者方面，丁晶最先提出将模糊相似选择应用于水文计算[15]；陈守煜在此基础上设计了相似流域选择的模糊集模型和算法[16]，弥补了传统水文比拟法自身不确定性的不足；邓红霞、李存军等提出基于集对分析（SPA）的相似流域选择方法[17]，通过联系度来描述所选流域间的同异程度；胡余忠、胡以宝等人利用流域几何特征测量方法[18]，基于DEM进行分水系的流域测算，研究流域间相似性；肖益民、梅汇梅提出基于灰关联的相似流域选择方法[19]，该方法与模糊集方法类似，都使用联系度表明两个集合间不确定的关系；李亚伟使用主成份分析方法（PCA）先对初始高维因素进行降维处理[20]，找出互不关联的主成份，将样本根据提取出的主成份属性用聚类方法测算相似度，最终得出相似流域，这是近年来广泛应用的方法。此外，将基于DEM提取的流域特征指标与年平均降雨量、水位、流量等水文要素结合在一起进行相似性分析有利于挖掘流域的水文规律。

**1.3本文工作**

本文的内容将从以下六个方面展开：

第1章 概述。主要介绍了基于DEM的相似流域查找系统的研究背景，国内外研究现状及相关技术的发展和应用情况，并对论文的组织结构进行了介绍。

第2章 基于DEM的数字流域特征及提取。本章简单介绍了DEM的定义及相关性质，并根据相似性分析理论需求对本文需要用到的基于DEM提取的数字流域特征进行了解释说明。同时介绍了数字流域特征的提取环境并以分水江的昌化站为例展示了数字流域特征的提取过程。

第3章 流域相似性分析的相关算法。本章详细介绍了本文需要用到的相似性分析算法，主要是对主成份分析、K-Means算法和层次聚类算法进行了学习，并用数据进行了实验分析，为系统开发提供理论支持。

第4章 基于DEM的相似流域查找系统的设计。本章介绍系统的需求分析、总体设计、数据库设计、功能模块、界面设计等，为系统实现提供模版。

第5章 基于DEM的相似流域查找系统的实现。本章简单介绍了系统的开发环境，对系统的实现进行了详细的说明，并展示了系统的操作流程。

第6章 本章对本论文完成的工作进行了总结，说明了论文所完成的研究，提出了不足之处以便以后的改进。

**第2章 基于DEM的数字流域特征及提取**

**2.1 DEM简介**

数字高程模型（Digital Elevation Model），简称DEM。它是用一组有序数值阵列形式表示地面高程的一种实体地面模型，是数字地形模型(Digital Terrain Model，简称DTM)的一个分支。数字地形模型是地形表面形态属性信息的数字表达，是带有空间位置特征和地形属性特征的数字描述，如水文下垫面、坡度、降雨、土壤类型等特征[22]。当DTM中地形属性为高程时，此时就是对地面高程信息的数字模拟，即为DEM。高程是空间坐标系中的第三维度坐标，数学表达为：z=f（x, y）。DEM的表达式如下所示,它处于三维坐标系中：

其中，（Xi，Yi）是而为空间位置，Zi是它们对应的高程值，当用规则有序的格网来表示地表的水平投影面时,DEM就可以简化为维度为1的向量集合：。按单元格网的连续性对其进行分类，可分为不连续、连续不光滑和光滑三种DEM。本文所采用DEM数据集为不连续DEM。其定义为数据结构单元表示了某一空间区域相同的属性特征信息，但数据值的变化只发生在相邻单元的交汇处的栅格数据集。本文中选用的DEM数据集是90×90分辨率数据。

**2.2 数字流域特征指标**

流域水文相似三要素由结构、驱动力和水流动力特征组成。其中将驱动力定义为外部能量和水分输入，结构要素包括植被、地形、土壤等因素[2]。当流域的下垫面条件发生显著改变时，流域径流、入渗、蒸发等水文过程同时也会改变，进而影响到水流的路径、流域的洪旱等问题。而降雨过程是制约流域地貌形成和改变的一大因素。依据流域下垫面特征和降雨量、流量等水文要素构建水文因子，对参证流域进行比较分析，是解决流域相似性问题的一个公认的可行方法。因此，本文基于DEM数据提取流域下垫面特征和流域结构特征值作为相似性分析的指标参数。

**2.2.1 数字流域特征相关概念**

（1）**流域：**流域的定义是河流某一断面的集水区域，由河系及坡地组成，如溪流、河流、湖泊和水库。

（2）**节点：**两条河流的交汇点。

（3）**河链：**河链是两点间的连接河段。可为河源和节点、出口和节点、临近节点间。当二叉树为N级时，外链和河源值为N，内链值为N-1，总值为2N-1。

河链分级共有5种不同的方法：Gravelius分级法、Horton分级法、Straher分级法、Shreve分级法和Scheidegger分级法。其中Shreve分级法规定：将尾端的无叉河流定义为1级，每当两个河流汇合，新汇合的河流级数等于汇集河流级数的代数和。

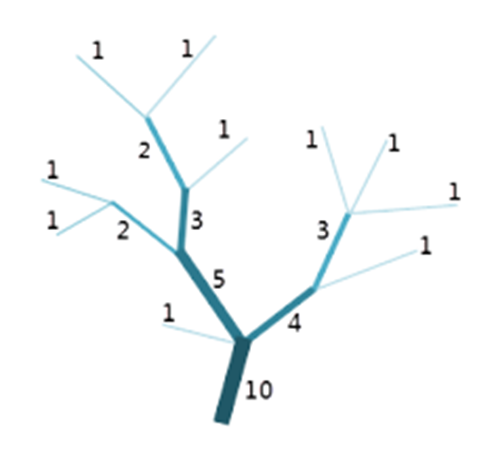


图2.1 shreve分级方法

（3）**流域高程曲线：**流域的高程曲线是指某高程的高程差值和它所对应的水平面的截面积之间的比例。一般用它们的相对值作图，即将水平截面积和流域面积的比值作为横坐标，将高程减去出口处高程再除以流域高差所得比值作为纵坐标。

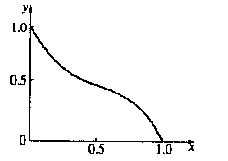


图2.2 流域高程曲线

其数学表达式为：



高程曲线的积分曲线为：



其中，y=h/H，X=a/A。h为出口断面之间的高程差，H为流域高度差值，a为h对应的水平截面面积，A为流域面积，a、d为常数，z为指数。

（4）**河长：**河长是指一条河流从河源沿着主流线到河口的平面投影长度。如果在一条河上沿主流方向截取上、下两个横断面，那么这两个横断面之间沿着主流线的平面投影长度就是其“河段长”[23]。

（5）**河道纵比降：**任意河段两端（水面或河底）的高差Δh称为落差，单位河长的高差称为河道纵比降。当河段纵断面近似垂直时，比降计算公式为：。其中J为河段比降，h1和h0为河段上、下断面高程，L为河段长度。

**2.2.2 基于DEM提取的数字流域特征**

数字高程模型(DEM)中蕴含着各种地形信息、气候信息、水文环境信息，随着电子技术的高速发展和提取算法的不断优化改进,提取的特征值种类和精度都有了较大的提高。本文基于DEM数据提取以下参数作为流域相似性分析的指标：

（1）**流域面积**：它表示流域的面积大小，即地表水的集水面积。它可以通过地图上两个区域间海拔最高点的连线——山脊线（分水线）确定，由分水岭所围的区域即为流域的范围，单位为km2。

（2）**流域长度：**流域中和主河道平行的最大长度值。

（3）**流域平均坡度：**将流域的地形图按照正方网格的形式划分，数量为100个及以上，依次计算出每个网格交叉点上与等高线正交方向的坡度，求取平均值。

（4）**形态因子：**流域面积与流域长度平方的比值称为形态因子。它和流域面积有关，流域面积越大，则它的值就越小，流域的形状向着狭长方面发展。形态因子直接反映流域的形状。

（5）**伸长比：**与流域面积等同的圆的直径Dc与流域长度L的比值称为伸长比Re，即：，该式还可表达为（A为流域面积），又因，可得。一般流域的伸长比在0.5到1.0之间。

（6）**河网密度：**单位面积含有的河流条数，单位为km/km^2，计算公式如下：



（7）**河道维持常数：**河网密度的倒数称为河道维持常数，又称水道给养面积，其含义是单位长度的河道必须要多少汇水面积来维持，单位为km^2/km。

（8）**平均河链长**：平均河链长等于水系总长度除以总河链数，计算公式为 。其中J为平均河链长，L为总长度，N1为河系中1级河流数，即河系的量级。

（9）**河链平均汇水面积**：其为流域总面积除以水系包含的河链数，单位为km^2。

（10）**河网总长度**：河网中所有河流的投影长度。

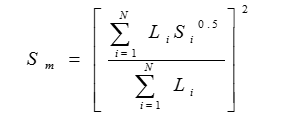
（11）**河流频度：**单位流域面积上的河流数，单位为1/km^2。

（12）**河链频度：**单位流域面积上的包含的Shreve分级的链数，单位为1/km^2。

（13）**常流性主河道长度**：一条树状水系是由许多河流交汇而成的，其中每条河流都有其级，每一级河流都有其长度，主河道长就是指水系中级数最高的河流从其河源沿着主流线到河口的平面投影长度。

（14）**常流性主河道比降**：划分主河道，根据定义求平均比降。

Johnstone和Cross方法：根据纵比降大致均匀的原则，将主河道划分成N个子河段。若第i个子河段的河长为Li，纵比降为Si，则主河道的平均纵比降由下式确定，



（15）**最大流路距离：**河流流过的总路程（非位移）。

（16）**流域高程曲线面积：**曲线和坐标系x、y轴所构成的图形的面积。

（17）**近似常数K：**又称无因次参数，其计算公式为 ，其中D为河网密度，J为平均河链长。

（18）**面积坡度：**根据流域高程曲线的定义，流域高程曲线的面积对面积－河长曲线的面积之比值，就是面积坡度。

**2.3 数字流域特征提取**

本节以江西省抚河流域为例介绍流域特征提取的原理和过程，其流域出口站为李家渡站。

**2.3.1 数据准备**

本文DEM数据源来自地理空间数据云，采用的是SRTM UTM 90米分辨率数字高程产品。SRTM数据全称为Shuttle Radar Topography Mission（航天飞机雷达地形测绘），由NASA和NIMA联合测量。其覆盖范围从60°N到56°S，达到全球陆地面积的百分之八十。SRTM地形产品按照分辨率分为30米精度和90米精度两种类型，其地理坐标系为WGS84。SRTM UTM 90 DEM是利用SRTM3 V4.1版本（由CIAT利用新的插值算法得到的SRTM地形数据）数据经过UTM（Universal Transverse Mercator Projection）投影系加工而来，它覆盖了整个中国区域。对以WGS84为空间度量的经纬度坐标进行UTM投影时，将经度加上180°除以六四舍五入得到的数值就是UTM投影区域的编号。

SRTM数据每5°经纬度方格划分一个文件，经度±180°分为72列，纬度±60°分为24行，文件名一般为utm\_srtm\_XX\_YY.IMG格式，XX表示列，YY表示行。李家渡站位于江西省进贤县李渡镇鉴良村，经纬度为116.1666667°E，28.21666667°N，计算可得DEM数据应取utm\_srtm\_60\_07.img。

**2.3.2 工具介绍**

目前，比较常用提取软件有：美国环境系统研究所(ESRI)研制的ArcGIS平台、RSI研制的River Tools软件和TOPAZ软件。ArcGIS在现今的研究中被广泛使用，其强大的功能、简洁的界面和具备二次开发能力颇受好评。整个ArcGIS Desktop平台包括ArcMap 、ArcCatalog、ArcGlobe和ArcScene。其中进行流域特征提取主要使用的是ArcMap、Arctool工具箱来完成操作。本文使用ArcGIS10.2平台使用DEM数据为基础对流域地表水文特征进行提取。

**2.3.3 提取步骤**

（1）安装ArcGIS10.2集成环境，打开ArcMap工具。

（2）建立站点Excel表格，将所选站点的名字、经度、纬度、类型信息写入Excel表格中，形式如下：

|  |  |  |  |
| --- | --- | --- | --- |
| Watersheds | Longitude | Latitude | type |
| 李家渡 | 116.1667 | 28.21667 | 2 |

一个站点建立成一个sheet表，且需要将文件保存为Excel 97-2003工作簿，后缀为xls的文件。

（3）将站点信息导入ArcMap。打开ArcMap文件→添加数据→添加XY数据，选择刚建立的站点sheet表，自动导入站点信息，选择WGS84作为输入坐标参考坐标系，点击确认。ArcMap会自动生成一个sheet事件点显示在图层中。



图2.3 站点导入

（4）站点投影。由于DEM数据格式为UTM\_SRTM，为了让站点能在DEM数据上显示，需要对其进行投影。打开ArcToolBox→数据管理工具→投影和变换→要素→投影，如图2.4所示：

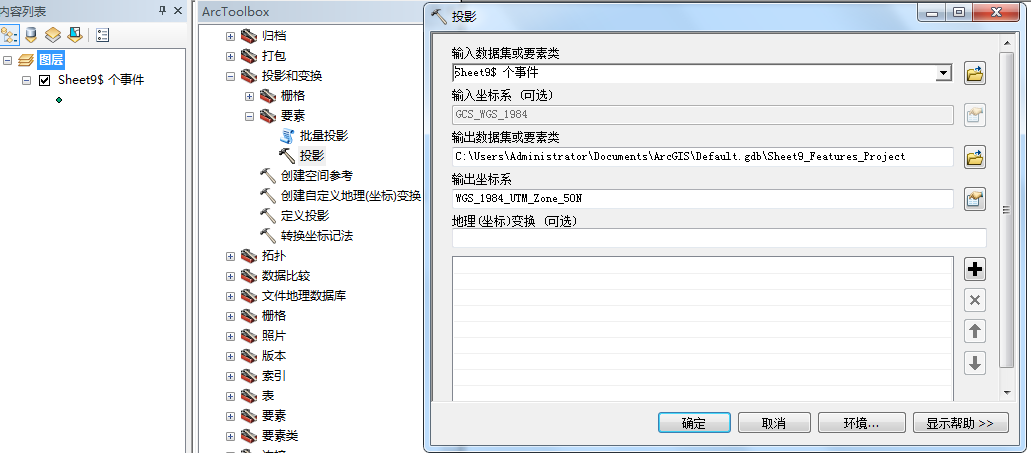


图2.4 站点投影

在输入数据集或要素类中选择刚建立sheet事件，输入或选择投影数据的存储位置。根据站点的经度加上180°的和除以6所得到的数据选择UTM投影区域，N代表北半球，也可以根据下载的DEM数据的空间参考系信息来选择。点击确认，ArcMap会自动生成投影点显示到图层中。

（5）将原始DEM数据导入ArcMap中。打开ArcMap文件→添加数据，选择所需的DEM高程数据，即utm\_srtm\_60\_07.img，点击添加导入到图层中。

（6）将DEM图层缩放到适当大小，重新截取数据框并命名，保存格式为TIF。



图2.5 截取数据框

（7）洼地填充。DEM是较光滑的地表模拟，但是因为内插值或者喀斯特地貌容易造成DEM表面存在一些凹陷的区域。这些区域会对水流方向的计算产生影响，并且在水系生成过程中易导致不合理的中断，降低数据处理的准确性。所以处理DEM第一步工作就是填充洼地。

打开ArcToolBox→Spatial Analyst 工具→水文分析→填洼。输入上一步重新提取的数据框utm\_srtm\_60\_071.tif，填写填洼后栅格的存储位置和名称，一般命名为fill.，栅格以tif格式存储。点击确定，自动生成数据并显示到图层中。



图2.6 填洼

（8）流向提取。流向指水流离开每个栅格单元时的指向，ArcToolBox的流向工具采用D8算法进行分析。首先，打开ArcToolBox→Spatial Analyst 工具→水文分析→流向，再输入填洼后的栅格数据集fill.tif，填写输出流向栅格数据的地址，一般命名为dir.tif。



图2.7 流向提取

（9）汇流分析。汇流累积量是根据水流的方向上游有多少个栅格最终流经该栅格的数量总和，其数值越大，该栅格代表的位置就更易形成地表径流。打开ArcToolBox→Spatial Analyst 工具→水文分析→流量。输入流向栅格数据dir.tif和权重栅格数据fill.tif，选择输出蓄积栅格数据文件地址，并将其命名为acc.tif。

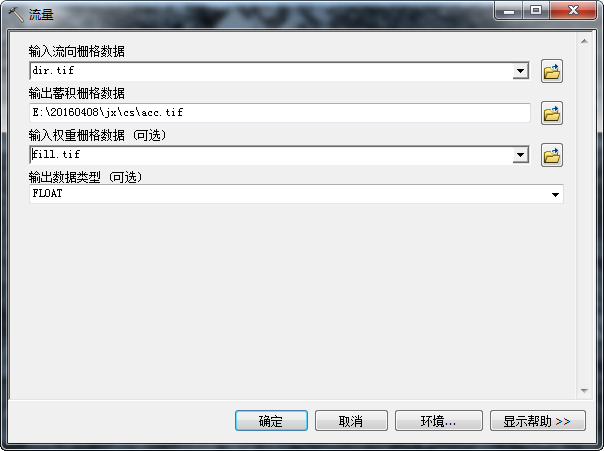


图2.8 汇流分析

（10）水系提取。基于栅格汇流累积量资料，依据河道最小集水面积提取水系。

打开ArcToolBox→HUDEM→水系提取，输入上一步骤中提取的汇流栅格数据acc.tif，根据需要输入河道集水面积阈值，选择水系数据的存储位置并将其命名为str.tif。阈值的选择影响到流域的大小，它表示河网中一点的最小集水量。阈值越小，河网密度则越大；反之，河网密度越小。本文定义阈值为0.5平方千米。



图2.9 水系提取

（11）确认站点位置。将李家渡站点位置缩放到水系数据所在的图层上，观察站点是否在代表水系的矢量格网上，点击编辑数据，适当微调点的位置，将其放在水系的汇流处，停止编辑进行保存。调整后的站点位置图如下：

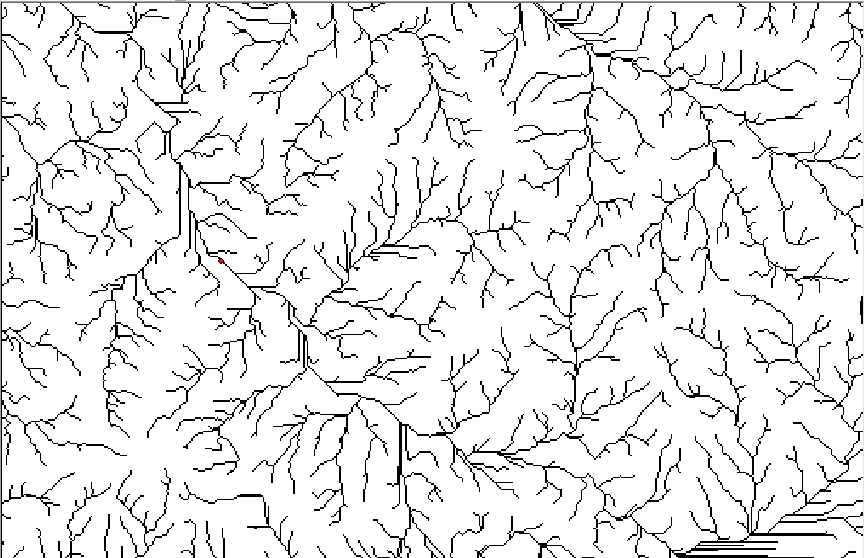


图2.10 站点位置确认

（12）流域提取。依据出口站点位置进行流域提取，当出口站点不在水系上时，对水系图层进行搜索并将距离出口站点最近的河道栅格作为出口。打开ArcToolBox→HUDEM→流域提取，输入已有的流向数据dir.tif，水系数据str.tif，李家渡站点投影，并将流域数据命名为basin.tif，填写其存储位置。

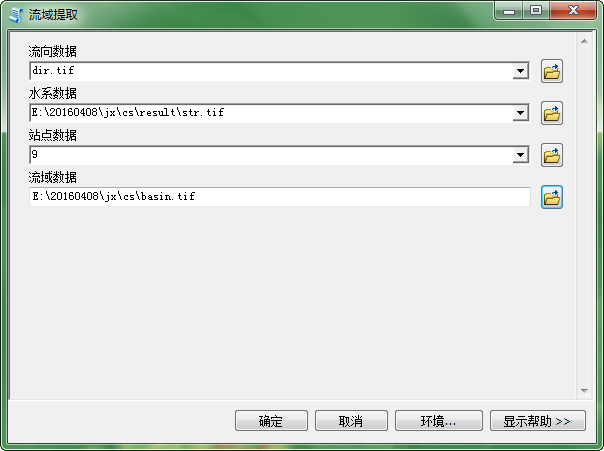


图2.12 流域提取

（13）计算流域坡度。打开ArcToolBox→HUDEM→流域平均坡度。输入栅格为流域数据basin.tif，将输出栅格命名为slope.tif并选择其存储位置。

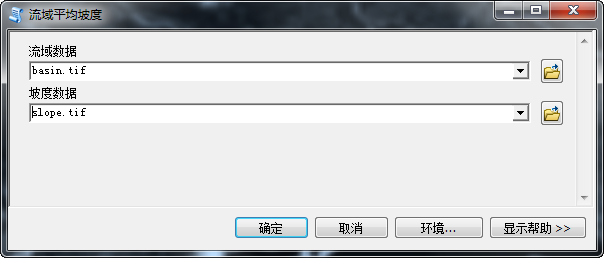


图2.11 坡度计算

（14）水系分级编码。按straler方法对提取水系进行分级统计。打开ArcToolBox→HUDEM→水系分级编码，输入流向数据dir.tif和水系数据str.tif，并将分级水系数据命名为order.tif，填写其存储位置。



图2.13 分级水系编码

（15）分级水系流域。依据分级水系和流向数据对流域进行划分。打开ArcToolBox→HUDEM→分级水系流域，输入流域数据basin.tif，流向数据dir.tif和上一步骤中的分级水系数据order.tif，并将分级水系流域数据命名为Orderbasin.tif，填写其存储位置。

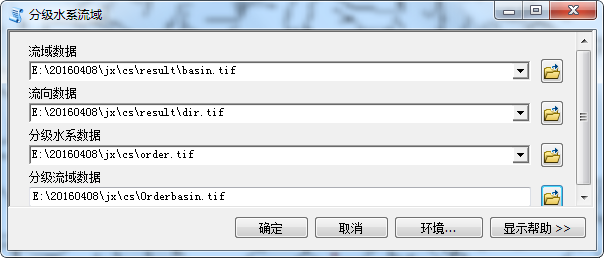


图2.14 分级水系流域

（16）流域距出口直线距离。统计流域内部各栅格区位距离流域出口直线距离。打开ArcToolBox→HUDEM→流域距出口直线距离，输入值为流域数据basin.tif，输出流域出口直线距离数据，将其命名为distance.tif。

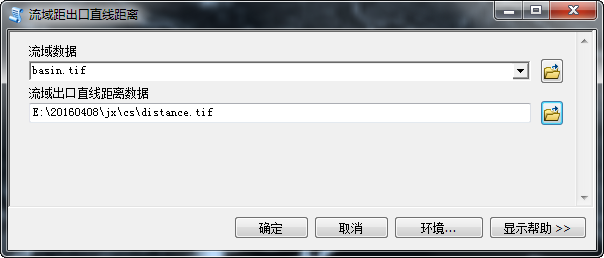


图2.15 流域距出口直线距离

（17）流域距出口流路距离。统计流域内部各栅格区位距离流域出口总体距离。打开ArcToolBox→HUDEM→流域距出口流路距离，输入值为流域数据basin.tif，流向数据dir.tif，输出流域距出口流路距离数据，将其命名为length.tif。

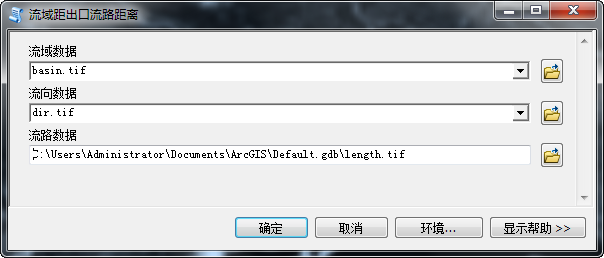


图2.16 流域距出口流路距离

（18）流域距水系流路距离。统计流域内部各栅格区位距离流域水系流路距离。打开ArcToolBox→HUDEM→流域距水系流路距离，输入值为流域数据basin.tif，流向数据dir.tif，和水系数据str.tif，输出流域距水系流路距离数据，将其命名为StrLength.tif。

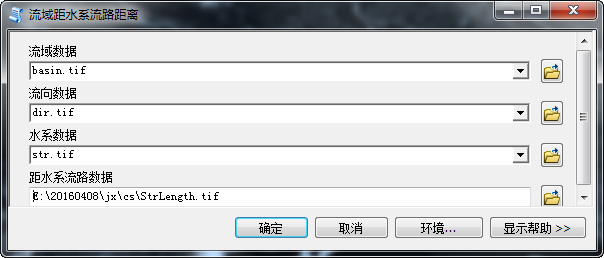


图2.17 流域距水系流路距离

（19）流域地貌参数统计。将从流域提取开始得到的数据整合分析，得到站点所属流域的流域特征值。打开ArcToolBox→HUDEM→流域地貌参数统计，输入数据包括流域数据basin.tif、流域高程数据utm\_srtm\_60\_071.tif、流域平均坡度数据slope.tif、流向数据dir.tif、水系数据str.tif、分级流域数据Orderbasin.tif、流域距出口流路距离length.tif、流域出口直线距离数剧distance.tif共八个栅格数据集。其输出为一个txt文件。

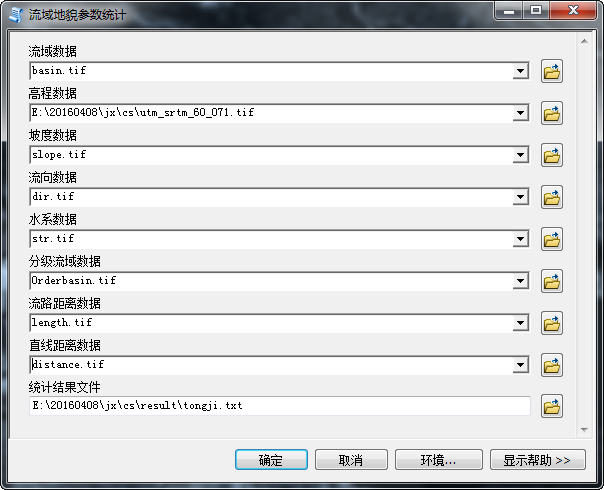


图2.18 流域地貌参数统计

李家渡站所在流域得到的统计文件如下图所示，包含本章介绍的18个流域特征值：

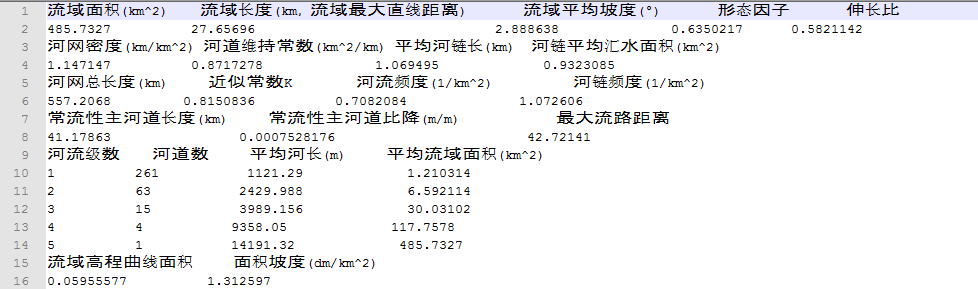


图2.19 流域特征参数

**2.4 小结**

本章简要介绍了DEM的相关概念，并从水文相似性分析的需求出发，详细介绍了基于数字高程模型能够提取的特征指标的定义及具体的提取步骤。为下文实现基于DEM的相似性流域查找提供了数据支持。

**第3章 流域相似性分析的相关算法**

数据挖掘（Data Mining），是通过计算机辅助来挖掘和分析大量数据，提取数据中规则和含义的过程。进入二十一世纪以来，国内数据库相关的技术手段得到了显著的提高。DEM空间数据步入大数据世代。数据库中的数据信息每天以千兆字节为单位迅速增长。但原始的DEM数据并不包含有效的流域特征信息，运用数据挖掘技术对原始DEM数据进行挖掘分析，建立系统的水文分析体系，是解决流域水文资料缺失的问题的基础。

流域相似的常用方法有以下几种：根据选择的随机性，运用PCA和聚类分析[21]的方法选择相似流域；根据选择的模糊性，使用模糊集的方法建立选择模型；灰色系统理论中的灰关联分析[20]。在这些算法中，本文选择聚类算法对流域参数指标进行分析比较。

**3.1 主成份分析**

主成分分析（Principal components analysis，PCA）是一种通过分析因素间协方差来进行降维的方法。将原始数据重新映射到一个新的基于数据方差的新坐标参考系。它将一系列可能相关的变量转化成一系列称为主成份的不相关的变量（数量会减少），通过保留低阶主成份、忽略高阶主成份来分析简化数据集。提取出来的主成份可作为后续分析预测中的变量。

主成份分析包括5个步骤：

（1）从每个数据维度减去所有数据维度的平均值，这样会产生一个平均值为零的数据集。

（2）计算协方差矩阵：,其中Cm×n是一个每个数据条目都是两个单独维度间计算协方差的结果。

（3）计算协方差矩阵的特征向量和特征值。

（4）选择成分，并形成一个特征向量：一旦发现协方差矩阵的特征向量，下一步是根据它们的特征值从高到低对它们进行排序。你选择的特征向量的数量将是新数据集的维度。这一步的目的是构建一个特征向量（矢量矩阵）。从特征向量的列表中选择需要的特征向量并形成一个矩阵列，如：

FV=(eig\_1,eig\_2, ..., eig\_n)。

（5）取FV的转置左乘原始数据集再转置。

**3.2 K-Means算法**

K-Means方法是一个基于原型的划分聚类技术，它根据聚类的重心将样本划分成用户指定数量的类别。它将一个数据对象的集合划分成互不重叠的子集（集群），每个数据对象是一个确切的子集。在K-Means方法中，每个集群与质心（中心点）相关联，并且每个点会被分配到距离最近的质心的集群。其中，聚类的个数必须指定，记为K。它是一个高效的贪婪算法，一个代表性的聚类算法。

K-Means算法基本步骤如下：

（1）定义初始质心，其个数为聚类数量K。可以随机分配所有组的质心也可以自己定义K个不同的值作为实体的质心。

（2）将每一个点分配给最接近的质心所在的类，为了找出最相似的质心，该算法必须计算所有样本到每个质心间的距离。

（3）重新计算质心的值。

（4）重复第二步和第三步，直至集群不再发生变化。

可以看出，初始质心的选择对后期聚类的精度影响很大，为减轻这种影响，当样本数量较少时，可以先使用其他方法求出特定的K个质心的值；当样本数量非常多时，随机选择了最初的质心后，后续质心的选择尽量远离初始质心的位置。

**3.3 层次聚类算法**

层次聚类将数据构造成树型结构（通常是二进制）。叶子是单独的数据项，而根是一个包含所有数据的单簇。树叶和根中间是包含数据子集的中间簇。层次聚类的主要思路是使“集群的集群”向上构建一棵树。 阶层式汇聚分群法（HAC）从底部开始，每个成员作为一个单簇，合并组合到一起，最后全部成员组成根，或者某个定义终结的条件被满足。分裂聚类是从一个大集团的所有数据开始，不断下分，直到每个成员作为一个单簇存在，或者满足终止条件。

假如有N个样本，层次聚类的基本过程如下：

（1）将每一个样本归为一个单独的类；

（2）成对计算所有类的距离，通过距离度量方法求取，如门式距离、欧式距离、马氏距离等；

（3）依据所得距离值构造距离矩阵；

（4）寻找距离最短的两个类；

（5）从矩阵中删除这两个类，并将它们合并成一个类；

（6）计算新合并的类到其他所有类的距离，并更新矩阵；

（7）重复上述步骤，直到距离矩阵被减少到一个单一的元素，或者达到了自定义的终止条件。

聚类分析算法的关键是怎样量化两个集群间的异同。层次聚类中类的合并方法有以下几种：

1. 最短距离法：又称单锁链准则，从所有测得的两个类之间的距离中选择相距最短的两个类进行合并。
2. 最长距离法：也称完全连锁聚类分析，和最短距离法相反，它是根据两个类间的最长距离进行类的选择合并。
3. 中间距离法：将两个类中的对象的距离求中间值，基于此中值进行合并。
4. 类平均法：将两个类中的对象的距离全部放在一起求平均值，基于此平均值进行合并。
5. 重心法：用比较所有类质心的差异来代替计算类间的距离作为合并标准。

层次聚类具有助于数据显示、分类细致的优点，但它没有提供能在早期阶段中“错误”对象的重定位，结果需要仔细核对以确保它符合条理。而且使用不同的距离公式测量簇集间距离时可能会产生不同的结果，建议进行多次实验并比较结果。

**3.4 相似流域分析实例**

本节以江西26个小流域为例，利用R语言进行流域相似性分析。

**3.4.1 数据准备**

原始数据中包括安和、大斜、东村、渡头、鹅公湾、高安、洪家园、居龙滩、李家渡、莲花、良田、麟潭、梅港、南关口、三都、上高、深渡、田头、万家埠、峡山、烟头、杨树坪、窑棚、宜丰、永修和樟树这26个流域出口站点的流域面积、流域长度、流域平均坡度、形态因子、伸长比、河网密度、河道维持常数、平均河链长、河链平均汇水面积、河网总长度、近似常数k、河流频度、河链频度、常流性主河道长度、常流性主河道比降、最大流路距离、流域高程曲线面积、面积坡度这18个基于DEM提取的流域特征值和1月平均降雨量、2月平均降雨量、3月平均降雨量、4月平均降雨量、5月平均降雨量、6月平均降雨量、7月平均降雨量、8月平均降雨量、9月平均降雨量、10月平均降雨量、11月平均降雨量、12月平均降雨量、多年平均降雨量这13个水文参数指标。

**3.4.2 数据分析**

将原始数据分别使用18个基于DEM提取的流域特征值作为特征因素（记为R1）进行分析。

1. 调用R语言的stats包对特征因素进行主成份分分析。

首先用fa.parallel()分析主成分个数,结果如下：R1经平行分析后建议取三个主成份。

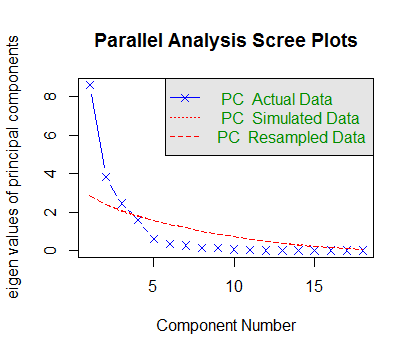


图3.1 R1主成份个数分析图

再使用principal函数对特征参数值进行降维,将R1的18个指标变换成3个独立的主成份因子。

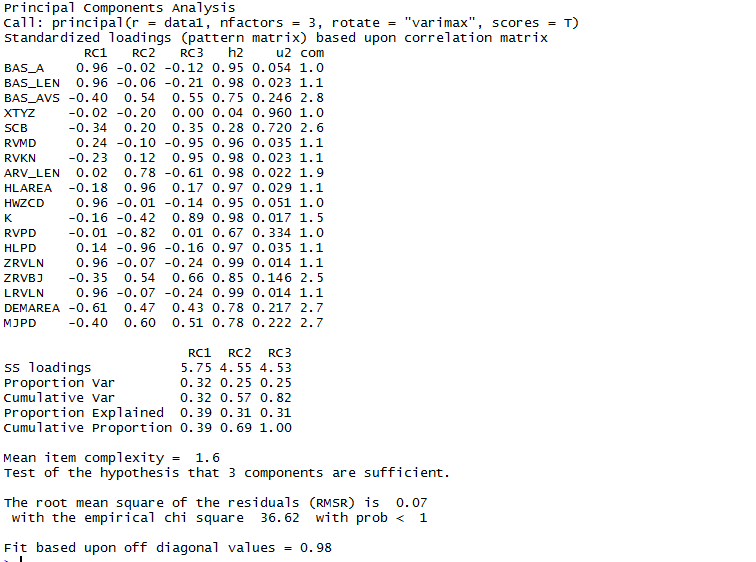


图3.3 R1主成份分析图

据图3.3所示，此处输入的是原始数据，并指定获取最大方差旋转的主成份，由于PCA值对相关系数矩阵进行分析，原始数据会自动被转换成相关系数矩阵，RC1-RC3栏包含了成分载荷（component loadings），表示因子与主成份间的相关系数。SS loading包含了主成份相关联的特征值，Proportin Var表示的是每个主成分对整个数据集的解释程度，第一主成分PC1解释了32%的方差，第二主成份PC2解释了25%的方差，第三主成份PC3解释了25%的方差。

1. 调用R语言中的层次聚类方法对经过主成份分析的数据进行分析。

表3.1 R1的三个主成份及其特征值

|  |  |  |  |
| --- | --- | --- | --- |
|  | 第1主成份 | 第2主成份 | 第3主成份 |
| 安和 | -0.81661 | 1.429826 | -0.49474 |
| 大斜 | -0.28889 | 0.621471 | 0.587337 |
| 东村 | 0.084263 | -0.79265 | -0.05008 |
| 渡头 | -0.96237 | -1.87727 | 0.431938 |
| 鹅公湾 | -0.67875 | -1.03855 | 0.212771 |
| 高安 | 0.005225 | -0.10227 | -0.71684 |
| 洪家园 | -0.06068 | 0.414479 | -1.08463 |
| 居龙滩 | -1.0824 | 0.062121 | -1.67467 |
| 李家渡 | 0.038787 | -0.44837 | -1.07827 |
| 莲花 | -0.29905 | 0.133916 | 0.325388 |
| 良田 | -0.68506 | -1.54256 | 0.927939 |
| 麟潭 | -0.43916 | 1.333999 | 1.839812 |
| 梅港 | -0.91306 | -0.10468 | -1.4895 |
| 南关口 | -0.65189 | -1.24403 | -0.53059 |
| 三都 | 0.311867 | 0.792925 | -0.65493 |
| 上高 | 0.795871 | -1.06032 | -0.4073 |
| 深渡 | 0.499737 | -0.22671 | 0.902744 |
| 田头 | -0.31695 | 0.920094 | 2.644591 |
| 万家埠 | 1.917708 | -0.03666 | 0.397815 |
| 峡山 | -1.04333 | 2.725364 | -0.68368 |
| 烟头 | -0.19657 | 0.364906 | 0.82093 |
| 杨树坪 | 0.264899 | -0.06064 | 0.253608 |
| 窑棚 | -0.25068 | -0.906 | 0.826998 |
| 宜丰 | -0.28754 | 0.050893 | -0.2453 |
| 永修 | 1.897787 | 0.316895 | -1.07454 |
| 樟树 | 3.156856 | 0.273841 | 0.013193 |

先计算数据间的距离，调用dist（）函数，method=“euclidean”，即欧几里德距离作为公式计算。然后，调用hclust()函数，分别使用其中的 “complete”、“single”、“average”三种距离方法对处理过的数据进行分析，画出聚类分析图。

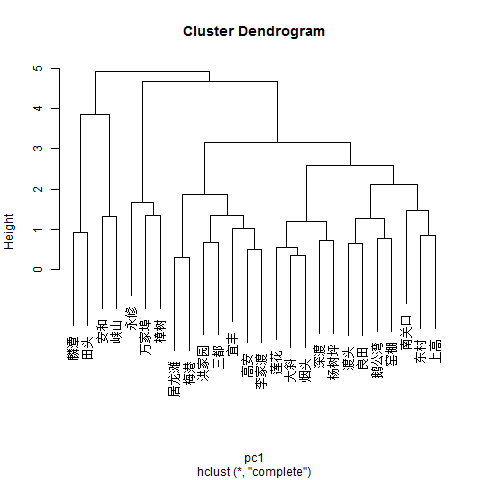


图3.4 最长距离法聚类分析

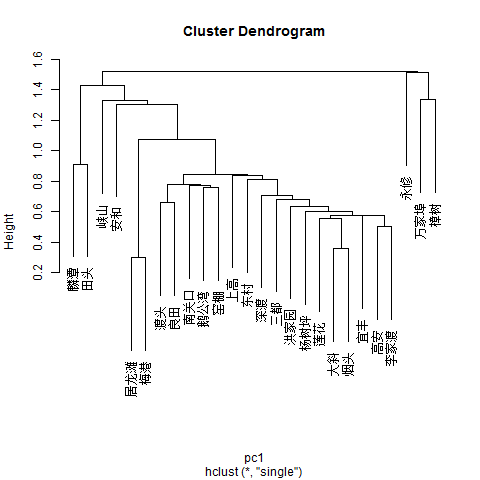


图3.5 最短距离法聚类分析

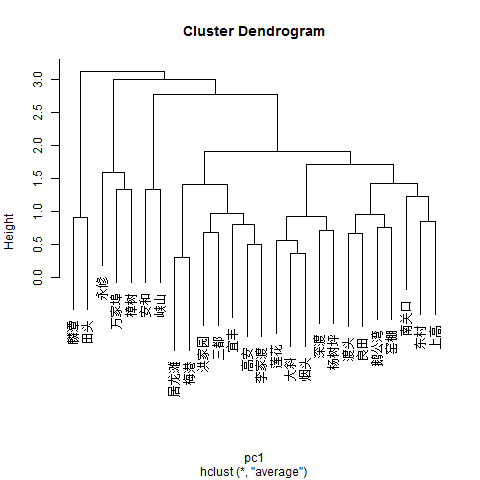


图3.6 平均距离法聚类分析

根据图3.4-3.6三张聚类结果图，我们可以发现万家埠、永修和樟树三个站在三种算法中都表现出较高的相似度，而南关口站在平均距离算法和最长距离算法中，和上高、东村两个站结果较为相似，但在最短距离算法中，它和窑棚、鹅公湾两个站结果较为相似。这表明层次聚类算法在使用不同的距离公式测量簇集间距离时可能会产生不同的结果，应多做几次实验进行比较。

1. 调用R语言Knn包中的K-Means方法对经过主成份分析的数据进行分析。

调用kmeans()函数，分别使用其中的 “MacQueen”、“Forgy”、“Lloyd”三种距离方法对处理过的数据进行分析，研究发现其结果中的规律。

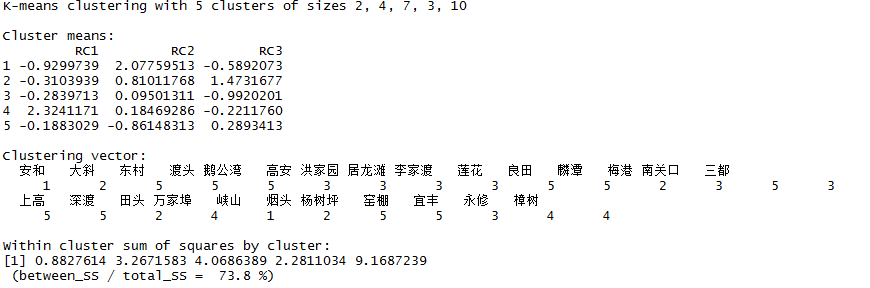


图3.7 Forgy方法

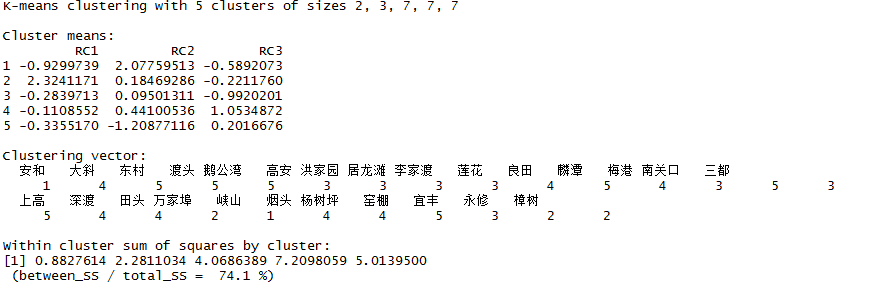


图3.8 MacQueen方法

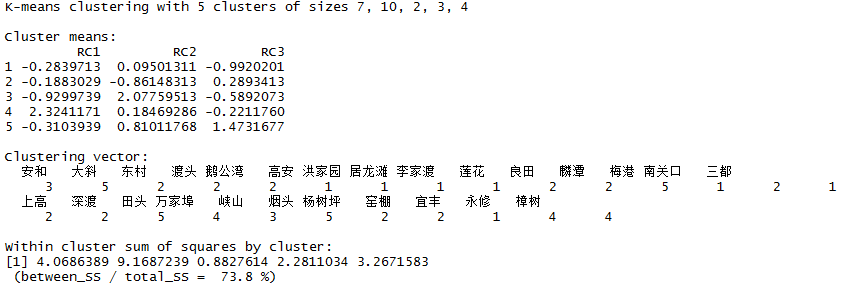


图3.9 Lloyd方法

Clustering vector表示每个样本所属的聚类，在一个聚类中的样本被认为是相似的，被划分成了一个类型。Within cluster sum of squares by clusters代表聚类内部的距离平方和。Between\_SS/total\_SS表示不同聚类间距离平方和及整体距离平方和的比值，比值越高，算法精确度越高。将上述结果进行比较发现，Forgy方法和Lloyd方法对26个站点的分类结果是相同的，但它们的聚类内部的距离平方和不相同，三个方法的精确值相差不大。

**3.5 小结**

本章主要对流域相似性分析的主成份分析步骤和两种基本算法进行了理论上的说明，并用江西省的26个出口站点信息作为数据进行了实例分析，对算法有了进一步的了解。为下文实现相似流域查找系统提供了技术支持。

**第4章 基于DEM的相似流域查找系统的设计**

我国幅员辽阔、地貌复杂，河域支流繁多，导致流域多样化。近年来，我国大江大河在洪水预报、流域治理方面取得了不错的成果。但众多的中小流域因流域资料不全或测站缺乏导致没有流域资料，在防洪防灾方面收效胜微。相似流域的查询对流域资料叠加整合、水文资料补全有着重要意义。因此，从数字高程模型提取有效特征参数，建立健全基础流域信息系统，整合共享有效信息对于中小流域治理有深远影响。本文基于两种相似性查询方法，设计并实现了基于DEM的相似流域查找系统。

**4.1 需求分析**

本文使用的数据分为两部分，一部分是从数字高程模型中提取到数字流域特征参数文件，格式为txt；另一部分来源于国家水文数据，数据类型涉及各个水文站的站码、经纬度、站名、行政区划以及雨量信息。本系统——基于DEM的相似流域查找系统的目标是：根据以上数据，通过一系列预处理之后，借助层次聚类、K-Means等算法，通过分析各个流域的水文数据，挖掘其中流域之间的相似性，并以图片的形式形象的展示给水利工作者。这些相似性信息能够帮助水利工作者分析水情，帮助水利工程设计者针对将要开展的水利工程寻找有效的参照，帮助流域监察者更好的处置突发情况。也为无资料或缺资料地区提供了更多能够参考的信息，对中小流域预防各种自然灾害有着重要意义。将来希望能够形成一个全国各流域信息的展示和比较平台，它的实现将能够给各类水利相关人员在处理问题时，提供相似流域针对相似问题的处理过程，以此作为参考，帮助其解决问题。

依据系统构建规划，本系统的主要功能包括：

（1）实现对流域特征数据以及DEM特征信息文件的增加和更新。

（2）实现对数据库中的流域信息数据的查询和预处理。

（3）实现对预处理结果，能够以两种不同的相似性分析方法进行分析。

（4）实现将分析结果以图表和文字等多种形式展现给用户。

**4.2 总体设计**

**4.2.1 系统结构设计**

基于DEM的流域相似性查找系统需要从数据库中读取数据进行分析并将分析结果以某种形式反馈给用户，适合基于J2EE平台的MVC模式来实现。MVC模式把用户与系统的交互分成三个不同的角色，各自独立处理自己的任务：Model处理应用程序中与数据相关的部分，通常负责在数据库中读写数据；View处理数据显示相关的部分，通常直接从模型中取得它需要显示的状态与数据；controller处于View和Model的中间，负责接受用户的输入，将输入进行分析并反馈给模型。根据上述要求，可将基于DEM的相似流域查找系统结构分为三个层次进行设计，如图所示：

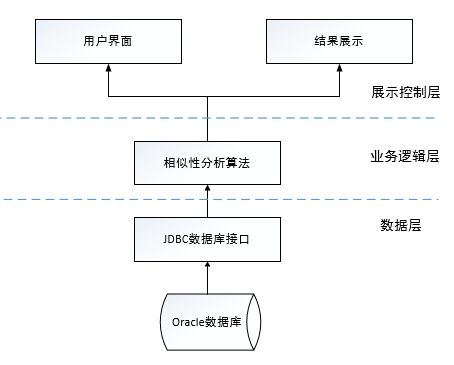


图4.1 系统层次结构图

1. 展示层: 其功能是接收用户提交的文件和输入的查询信息等前台输入，当后台处理分析后，处理结果会返回该层被展示，由Jsp和JavaScript组件实现。
2. 控制层：其功能是接受并处理从展示层收到的任务信息，并调用业务逻辑层的代码进行实现。由Servlet组件实现。
3. 业务逻辑层：根据Servlet发来的指令，从数据层的数据库中获取所需数据，进行分析，得到结果后，将其反馈给表示层进行展示，由相似性分析算法组件构成。
4. 数据层：根据业务层发送的请求读取数据到业务层中，由数据库和数据接口组成。

**4.4.2 系统流程设计**

系统运行是一个接收、处理、完成用户请求的循环过程。这期间，对于用户发布的不同类型的请求，系统会加载相应的模块去解决。系统整体流程设计如下图所示。

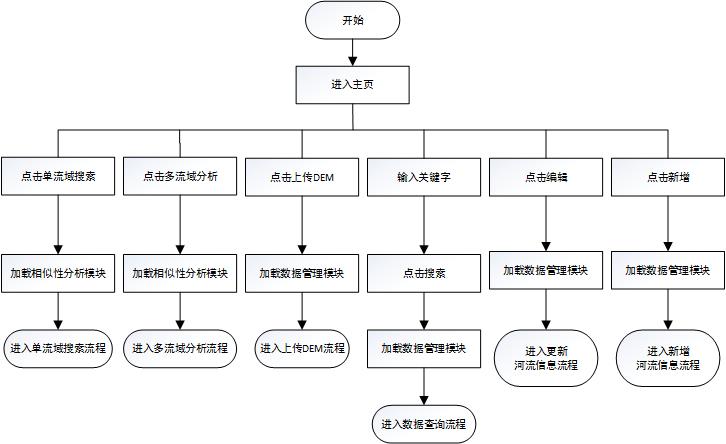


图4.2 系统流程图

**4.3 系统功能设计**

基于DEM的相似流域查找系统是一个深度挖掘DEM有效信息，查询分析河流水文信息的系统。本系统主要包括相似性分析和数据管理两个功能模块。其中相似性分析模块分为单流域搜索和多流域分析两种方式。而数据管理模块可以添加、更新和查询河流信息和上传DEM参数文件。基于DEM的相似流域查找系统功能结构如下图4.3所示。



图4.3 系统功能结构图

**4.4数据库设计**

基于DEM的相似流域查找系统数据来源于国家水文数据库和DEM高程数据。由于全国的水文数据量庞大，需要一个相对稳定的存储环境，而Oracle在安全性、稳定性方面优于其他数据库，且Oracle在数据导入和数据多种格式支持方面功能强大，所以选择Oracle 11g作为数据存储的工具。系统涉及到以下两张表：河流相似性分析基本信息表和河流相似性性分析属性字典表。

**4.4.1河流相似性分析基本信息表**

表标识为SIMILAR\_DATA，该表主要存储河流基本信息和流域特征信息。内容如下表所示：

表4.1 河流相似性分析基本信息表主要表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 标识符 | 类型及长度 | 有无空值 | 计量单位 | 主键 |
| 1 | 河流编码 | RVCODOE | nvarchar(12) | N |  | Y |
| 2 | 河流名称 | RVNAME | nvarchar(100) | N |  | Y |
| 3 | 行政区划 | ADDVCD | nvarchar(100) | N |  |  |
| 4 | DEM状态 | DEMFLAG | nvarchar(10) | N |  |  |
| 5 | 时间戳 | TIMESTAMP | nvarchar(20) | N |  |  |
| 6 | 类型 | TYPE | nvarchar(20) |  |  |  |
| 7 | 流域面积 | BAS\_A | number(8,2) |  | km^2 |  |
| 8 | 流域长度 | BAS\_LEN | number(9,4) |  | km |  |
| 9 | 流域平均坡度 | BAS\_AVS | number(15,8) |  | ° |  |
| 10 | 形态因子 | XTYZ | number(8,4) |  |  |  |
| 11 | 伸长比 | SCB | number(8,4) |  |  |  |
| 12 | 河网密度 | RVMD | number(8,4) |  | km/km^2 |  |
| 13 | 河道维持常数 | RVKN | number(8,4) |  | km^2/km |  |
| 14 | 平均河链长 | ARV\_LEN | number(8,4) |  | km |  |
| 15 | 河链平均汇水面积 | HLAREA | number(9,4) |  | km^2 |  |
| 16 | 河网总长度 | HWZCD | number(8,2) |  | km |  |
| 17 | 近似常数k | K | number(8,4) |  |  |  |
| 18 | 河流频度 | RVPD | number(8,4) |  | 1/km^2 |  |
| 19 | 河链频度 | HLPD | number(8,4) |  | 1/km^2 |  |
| 20 | 常流性主河道长度 | ZRVLN | number(8,2) |  | km |  |
| 21 | 常流性主河道比降 | ZRVBJ | number(8,4) |  | m/m |  |
| 22 | 最大流路距离 | LRVLN | number(8,2) |  |  |  |
| 23 | 流域高程曲线面积 | DEMAREA | number(8,4) |  |  |  |
| 24 | 面积坡度 | MJPD | number(8,4) |  | dm/km^2 |  |
| 25 | 1月平均降雨量 | JARAVP | number(8,2) |  |  |  |
| 26 | 2月平均降雨量 | FEBAVP | number(8,2) |  |  |  |
| 27 | 3月平均降雨量 | MARAVP | number(8,2) |  |  |  |
| 28 | 4月平均降雨量 | APRAVP | number(8,2) |  |  |  |
| 29 | 5月平均降雨量 | MAYAVP | number(8,2) |  |  |  |
| 30 | 6月平均降雨量 | JUNAVP | number(8,2) |  |  |  |
| 31 | 7月平均降雨量 | JULAVP | number(8,2) |  |  |  |
| 32 | 8月平均降雨量 | AUGAVP | number(8,2) |  |  |  |
| 33 | 9月平均降雨量 | SEPAVP | number(8,2) |  |  |  |
| 34 | 10月平均降雨量 | OCTAVP | number(8,2) |  |  |  |
| 35 | 11月平均降雨量 | NOVAVP | number(8,2) |  |  |  |
| 36 | 12月平均降雨量 | DECAVP | number(8,2) |  |  |  |
| 37 | 多年平均降雨量 | MYAVP | number(8,2) |  |  |  |

1）河流编码：按《国家水资源监控能力建设项目标准——信息分类及编码规定》4.1 节执行。

2）河流名称：河流编码对应河流的名称。

3）行政区划：是国家为了进行分级管理而实行的区域划分，按照《[宪法](http://baike.baidu.com/view/3575.htm)》规定中国行政区划分为省、县、乡三级，这里指的是河流所属的行政区划。

4）DEM状态：标明DEM数据是未上传状态还是已上传状态。

5）时间戳：此记录第一次录入数据库的时间或者数据更新时间，取系统日期时间。

6）类型：指流域的大小情况，如中型流域、小型流域。

7）—24）为DEM数据中提取的属性值。

25）—37）为水文相关属性值。

**4.4.2河流相似性性分析属性字典表**

表标识为SIMILAR\_INDEX，该表是一张字典表，主要对河流相似性分析基本信息表SIMILAR\_DATA中的字段信息进行说明，对SIMILAR\_DATA中属性在前台显示的顺序进行定义。内容如下表所示：

表4.2河流相似性性分析属性字典表主要表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 字段名 | 标识符 | 类型及长度 | 有无空值 | 计量单位 | 主键 |
| 1 | 属性标识符 | CODE | NVARCHAR2(10) | N |  | Y |
| 2 | 属性名称 | NAME | NVARCHAR2(10) |  |  |  |
| 3 | 数据单位信息 | UNIT | NVARCHAR2(10) |  |  |  |
| 4 | 类型 | TYPE | NVARCHAR2(10) |  |  |  |
| 5 | 前台显示顺序 | ORDERBY | NVARCHAR2(2) |  |  |  |

**4.5详细设计**

**4.5.1 单流域搜索**

单流域搜索应实现如下功能：

1. 页面展示数据库中存储的河流信息；
2. 选出一条河流作为基准河流，这条河流的选择范围应该是数据库表存储的全部河流信息，基准河流可从河流数据表中勾选；
3. 与基准河流比较的河流的数量不限；
4. 因不能在一页中完全显示数据库中所有河流信息，河流信息表应分页显示；
5. 当选择的待比较的河流数量较多，而用户想要更新比较范围时，一键重置比较范围，将之前所选河流清空；
6. 当用户选择错少数待比较河流时，可以移除错误河流；
7. 因相似性指标含有18个DEM要素和13个水文要素，相对较多，用户可以选择自由组合这些要素进行相似性分析；
8. 可以按照行政区划显示处于该行政区的河流信息；
9. 用户可以自己调整Kmeans的聚类数目，比较不同的聚类数目下分析结果的异同；
10. 分析结果以图形、表格、文字等形式在前台界面展示。

单流域搜索的流程如下图所示：



图4.4 单流域搜索流程图

**4.5.2 多流域分析**

多流域分析应实现如下功能：

1. 页面展示数据库中存储的河流信息；
2. 待分析的河流数量不限，河流的选择范围应是数据库表存储的全部河流信息，可从河流数据表中勾选；
3. 因不能在一页中完全显示数据库中所有河流信息，河流信息表应分页显示；
4. 当选择的待分析的河流数量较多，而用户想要更新全部范围时，一键重置，将之前所选河流清空；
5. 用户能够移除已选的河流；
6. 用户可以自由选择组合相似性分析要素进行分析；
7. 可以按照行政区划显示处于该行政区的河流信息；
8. 用户可以自己调整聚类数目，比较不同的聚类数目下分析结果的异同；
9. 分析结果以图形、表格、文字等形式在前台界面展示。

多流域分析流程如下图所示：



图4.5 多流域分析流程图

**4.5.3 DEM参数文件上传**

当一些河流缺少DEM参数信息时，用户可以按照系统提供的文件格式上传DEM参数文件。因按照本文第二章所述提取DEM流域特征方法得到的DEM参数文件中除去本文使用的18个DEM特征指标外，还有归一化面积、归一化高程等大量其它数据。用户上传DEM参数文件后，先保存在服务器上，系统需要对文件进行分析，将系统所需要的参数提取出来，存入数据库中。因上传文件的名字并不统一，为了方便将提取的流域的DEM参数与河流信息匹配，把文件的名字命名为河流代码。

DEM参数文件上传流程如下图所示：



图4.6 DEM参数文件上传流程图

**4.5.4 查询河流信息**

输入河流代码或者河流名字等关键词，能在河流信息表中查询到相对应的河流信息。

**4.5.5 更新河流信息**

界面上应该给出一个编辑按钮，当点击编辑按钮时会弹出更新信息的界面。选择想要更新的河流，点选编辑按钮，输入修改的河流信息后确认更新。

**4.5.6 新增河流信息**

界面上应该给出一个新增按钮，当点击按钮时会弹出新增数据的界面。新增数据的界面应该包括河流编码、河流名称、行政区划、DEM状态、时间戳、类型、流域面积、流域长度、流域平均坡度、形态因子、伸长比、河网密度、河道维持常数、平均河链长、河链平均汇水面积、河网总长度、近似常数k、河流频度、河链频度、常流性主河道长度、常流性主河道比降、最大流路距离、流域高程曲线面积、面积坡度、1-12月平均雨量和多年平均雨量这些属性信息的输入框。

**4.5.7 界面设计**

基于DEM的相似流域查找系统查询方法界面UI设计如图4.7所示，顶部为“系统logo区”，其下方是“相似性查询功能选择区”，左上部分是“行政区划列表”，左下部分则是“河流范围选择区”，右半部分主要是“数据显示及结果展示区”和“相似性要素选择区”。其中“相似性要素选择区”是一个下拉框。



图4.7 界面设计图

1. 系统logo区：基于DEM的相似流域查找系统的名字和标识图片；
2. 相似性查询功能选择区：选择单流域搜索功能和多流域分析功能；
3. 行政区划表：对全国的城市按省→市→县（区）的级别进行分级显示，点击该城市，数据展示页面就会显示该城市的河流信息；
4. 河流范围选择区：单流域搜索在这个区域确定待搜河流和搜索范围，多流域分析在这个区域确定河流分析范围，即需要分析的全部河流在这个区域确定；
5. 相似性要素选择区：选择需要的相似性要素，共31个可供选择，包括18个DEM特征参数和13个水文要素；
6. 数据显示及结果展示区：以表格的形式展示河流信息数据，在查询结果返回后以图形和表格的形式进行展现。

**4.6 小结**

本章主要对基于DEM的相似流域查找系统做了详细的设计，首先对系统的需求进行了分析，并提出了系统的总体设计。然后对系统功能进行了分析讨论、设计了数据库表，最后基于上述工作，对系统进行了详细设计。

**第5章 基于DEM的相似流域查找系统的实现**

**5.1 系统开发环境**

* 操作系统：Windows 7
* 集成开发环境：ideaIU-13.1.6
* JAVA版本：JDK 1.7，JAVA EE 1.7
* Web 应用服务器：Tomcat-7.0.68
* 浏览器：IE 11，Firefox 11.0，Chrome
* 数据库：ORACLE 11g

**5.2 系统实现**

**5.2.1 单流域搜索功能实现**

根据上一章对单流域搜索模块的设计，当用户从前台页面选择了基准河流和与其进行比较的河流范围，确定了需要的相似性分析要素，点击确定后，后台的DataQueryServlet会根据前台传过来的一个flag进行判断。

当该flag的值为"singlerv"时，从前台页面获取singleCode、singleListCodes、checkboxsCodes、k四个参数，它们分别代表了待搜流域、待搜范围、要素范围和聚类数目。然后根据这些参数信息，调用SimilarResult类中的getMultiRes( )方法和ClusterDao类中的getclusterData( )方法从数据库中取出聚类需要用的数据。控制台再调用R语言的Kmeans方法进行分析,并将结果返回前台页面。其中，使用R语言调用Kmeans聚类算法是本节实现的重点。

上述过程中使用到以下几个类，及它们之间的关系如下图所示：

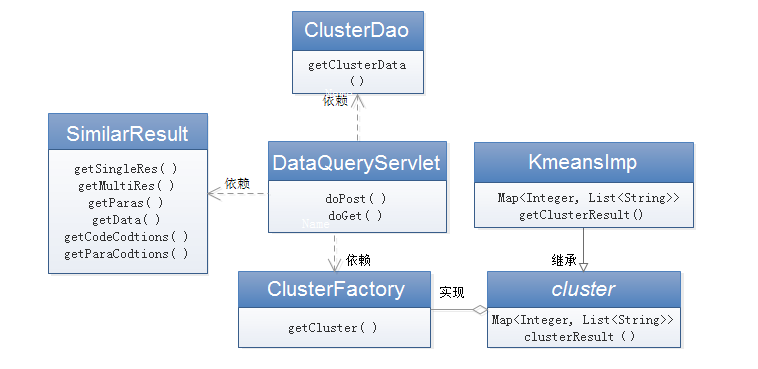


图5.1 单流域搜索uml类图

DataQueryServlet：控制台，对前台传来的flag判断类型后，调用各种类实现操作；

ClustDao：用于从数据库中获取分析需要的河流信息；

Cluster：调用R语言分析方法的接口；

ClusterFactory：实现Cluster的类；

KmeansImp：cluster的子类，调用R语言中Kmeans方法的具体代码。

本系统先连接Rserve服务器，向服务器发送需要分析的数据，调用写好的R语言脚本进行分析后，Rserve服务器将结果返回给控制台。JAVA调用Kmeans方法的脚本如下：

|  |
| --- |
| RserverConf rconf = new RserverConf("127.0.0.1", 6311, "conan", "conan", new Properties());  Rsession s = Rsession.newInstanceTry(System.out, rconf);  // 从文件加载R环境  String filePath = "E:/R/code/inputK.R";  s.eval("source(\"" + filePath + "\",encoding = 'UTF-8')"); |

K-means聚类过程中先要对相似性要素进行主成份分析，调用R语言psych包中的方法进行实现，代码如下：

|  |
| --- |
| #PCA降维  pc<-principal(basins, nfactors = k, score = T, rotate = "varimax")  means <- apply(basins,2,mean)  for (i in 1:ncol(basins)) {  if(i == 1){  basinsNew = data.frame(basins[,1] - means[1])  }else{  basinsNew = data.frame(basinsNew,basins[,i] - means[i])  }  names(basinsNew)[i] = names(basins)[i]  }  x<-as.matrix(basinsNew)  y<-pc$loadings  mainBasins <- x%\*% |

再进行K-means聚类分析：

|  |
| --- |
| fit\_kml <<- kmeans(mainBasins,center = k)  betweenss <<- fit\_kml$betweenss  totss <<- fit\_kml$totss #找出selectedCountry在countries中的位置  selectedRvindex=which(Basins$RVCodes == selectedrv)  #找出selectedCountry在cluster中的类别  cluster\_aim=fit\_kml$cluster[selectedRvindex]  #找出同属于cluster\_aim所有河流的index  similarityIndexs=which(fit\_kml$cluster == cluster\_aim)  #查找到的相似的河流  similarRvs <<- Basins[similarityIndexs,] |

因结果维度较高，当以图片的形态展示分析结果时，经MDS降维后，运用ggplot包中的方法画图并保存。

|  |
| --- |
| mds = cmdscale(dist(mainBasins,method = 'euclidean'),k=k,eig=T)  x = mds$points[,1] #MDS降维  y = mds$points[,2]  p <- ggplot(data.frame(x,y),aes(x,y,label=c(RVNames)))#画图 |

**5.2.2 多流域分析功能实现**

多流域分析功能的实现与单流域搜索的过程相似，都是通过调用R语言中的分析方法进行实现，其实现过程中用到的类和它们的关系如下图所示：

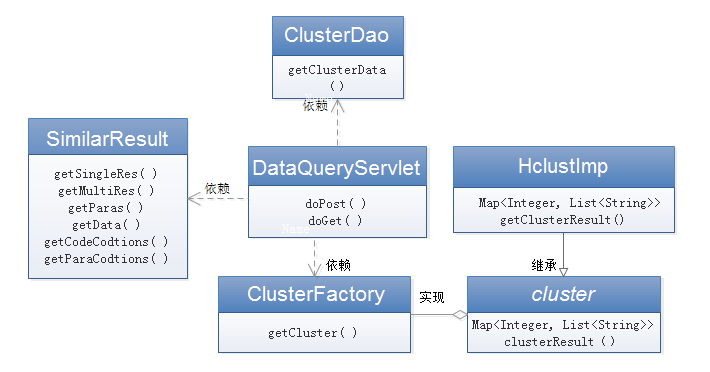


图5.2 多流域分析uml类图

DataQueryServlet进行判断并获取前台传输参数的代码如下：

|  |
| --- |
| if (flag.equals("multirv")) {  String multiListCodes = request.getParameter("multiListCodes");  String checkboxsCodes = request.getParameter("checkboxsCodes");  String kStr = request.getParameter("k");  String method = request.getParameter("method"); |

这里从前台得到的参数分别是multiListCodes多流域搜索的范围、checkboxCodes相似性分析参数、k聚类数目和聚类方法。然后根据这些参数信息，调用SimilarResult类中的getMultiRes( )方法和ClusterDao类中的getclusterData( )方法从数据库中取出聚类需要用的数据。控制台再调用R语言的方法进行分析,并将结果返回前台页面。

层次聚类算法的实现代码如下：

|  |
| --- |
| compoents <- fa.parallel(basins, fa = "pc", n.iter = 100)  compoentNumber <- compoents$ncomp  dist.r <- dist(mainBasins,method = 'euclidean')  #运用欧几里德距离求样本间的距离  fit\_hc <<- hclust(dist.r)  #进行聚类分析  result <<- cutree(fit\_hc,k=k)  最后返回聚类结果。 |

**5.2.3 上传DEM参数文件功能实现**

上传DEM参数文件功能实现过程中涉及到三个类，分别是用来控制DEM参数文件上传过程的UploadTxtServlet，用其中updateDemFlag()来更新Dem状态的EditUtil类和用来解析DEM文件的ExtractDem类。它们之间的关系如下图所示：

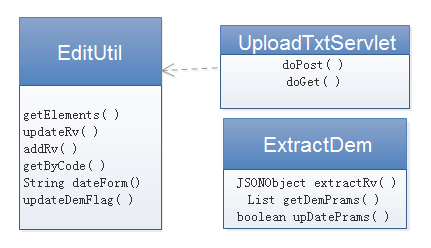


图5.3 DEM文件上传uml类图

首先UploadTxtServlet能从前台表单获取到DEM参数文件在本地的存储路径和它相对应的河流代码这两个参数，当接收完文件后，系统会将它保存在服务器上的一个指定文件夹中，确保文件的安全性，并将文件的名称改成河流代码。实现代码如下：

|  |
| --- |
| public void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  //上传DEM的河流编码  String rvcode=null;  //得到上传文件的保存目录，将上传的文件存放于WEB-INF目录下，不允许外界直接访问，保证上传文件的安全  String savePath =this.getServletContext().getRealPath("/uploadfiles");  String saveFileName = "temp";  File file = new File(savePath);  //判断上传文件的保存目录是否存在  if (!file.exists() && !file.isDirectory()) {  System.out.println(savePath+"目录不存在，需要创建");  file.mkdir(); //创建目录  } |

系统使用Apache的commons包中的文件上传组件来传输文件：创建一个DiskFileItemFactory工厂；创建一个文件上传解析器；解决上传文件名的中文乱码；使用ServletFileUpload解析器解析上传数据，解析结果返回的是一个List<FileItem>集合，每一个FileItem对应一个Form表单的输入项。代码如下：

|  |
| --- |
| DiskFileItemFactory factory = new DiskFileItemFactory();  ServletFileUpload upload = new ServletFileUpload(factory);  upload.setHeaderEncoding("UTF-8");  List<FileItem> list = upload.parseRequest(request); |

因为上传的DEM参数文件为txt格式，里面除了包含了本文需要的18个DEM流域特征参数，还有其他信息。所以我们要将需要的参数从文本中提取出来放到数据库中，使用的是ExtractDem类。其中上传文件的格式如图所示：

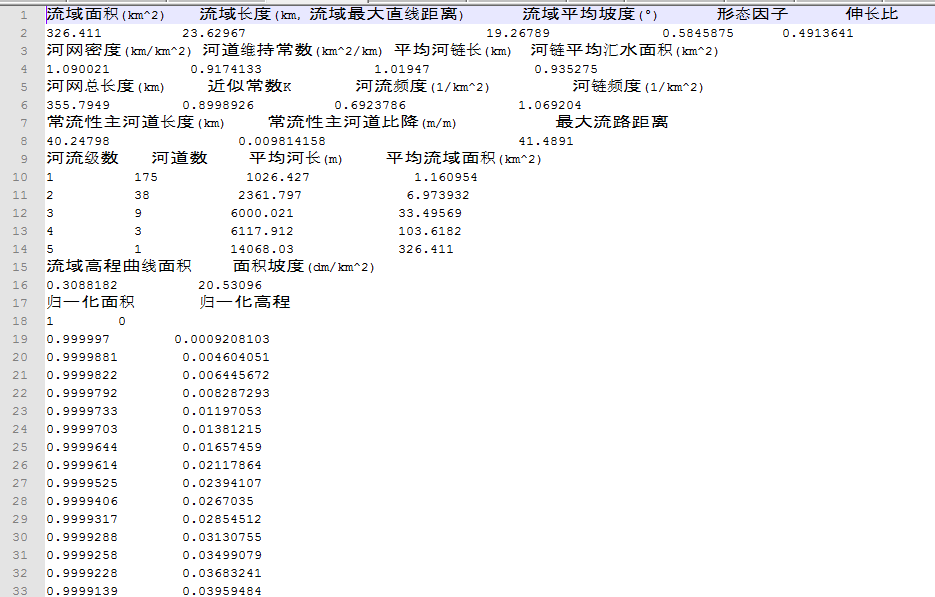


图5.4 文件格式

可以看出，txt格式并不规则，且包含其他特征信息。在ExtractDem类中，按行读取txt文件，直接取得前八行中第二、四、六、八行的值至对应的属性，当行数大于九时，设定一个flag=0，当读到值为“流域高程曲线面积”时，令flag等于该行的数值加一，取flag位置的数值到流域高程曲线面积和面积坡度属性值中。

|  |
| --- |
| JSONObject extractRv(String rvcode,HttpServletRequest request){  JSONObject jsonObject=new JSONObject();  List DemPrams=getDemPrams();  List DemPramsList=new ArrayList();  String message="success";  // 标记“流域高程曲线面积”“面积坡度” 两个参数的位置  if(file.exists()&&file.isFile()){  List lines=file.readLines();  int count=0;  for(String line:lines){  count++;  switch (count){  case 2:  String[] line2=line.split(" ");  for(int i=0;i<line2.length;i++){  if(!line2[i].equals("")){  DemPramsList.add(DemPrams.get(pramIndex)+"="+line2[i]);  pramIndex++;  }  } //获取上传文件目录  String savePath = request.getServletContext().getRealPath("/uploadfiles");  String fileName=rvcode+".txt";  File file = new File(savePath+"/"+fileName);  int pramIndex=0; //下dian面参数顺序  int Flag=0; //  break;  case 4:  String[] line4=line.split(" ");  for(int i=0;i<line4.length;i++){  if(!line4[i].equals("")){  DemPramsList.add(DemPrams.get(pramIndex)+"="+line4[i]);  pramIndex++;  }  }  break;  case 6:  String[] line6=line.split(" ");  for(int i=0;i<line6.length;i++){  if(!line6[i].equals("")){  DemPramsList.add(DemPrams.get(pramIndex)+"="+line6[i]);  pramIndex++;  }  }  break;  case 8:  String[] line8=line.split(" ");  for(int i=0;i<line8.length;i++){  if(!line8[i].equals("")){  DemPramsList.add(DemPrams.get(pramIndex)+"="+line8[i]);  pramIndex++;  }  }  break;  } |

**5.2.4 河流信息管理功能实现**

本系统提供河流信息的新增、更改和查询功能。这三个功能都是依赖DataQueryServlet和EditUtil两个类实现。两个类之间的关系如下所示。

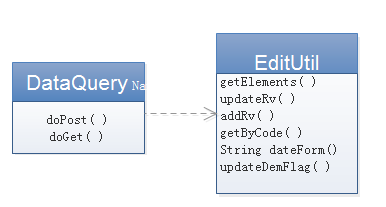


图5.5 UML类图

其中，updateRv()是实现更新河流的方法，addRv()是实现新增河流的方法，getByCode()是实现根据河流编码查询河流信息的方法。

**5.2.5 系统界面实现**

根据上一章的设计，系统界面中的行政区划列表是一个树形结构的选择枝，对全国的城市按省→市→县（区）的级别进行分级显示。通过应用ExtJS框架来实现。ExtJS是一个和后台不相关的，主要作用是新建和美化前端用户界面的框架。它使用javascript语言编写。代码实现如下：

|  |
| --- |
| Ext.onReady(function(){  var addvcdTree = new Ext.tree.TreePanel({  id:'mytree',  renderTo: 'addvcdtree',  title:"行政区划",  height:800,  collapsible:true,  useArrows: true,  autoScroll: true,  containerScroll: false,  border: false,  rootVisible:false,  root: new Ext.tree.AsyncTreeNode({  id:"9999",  text:"根节点",  lines:true,  loader: new Ext.tree.TreeLoader({  dataUrl:"../TreeServlet?method=addvcd&id="+"0"  })  }),  listeners:{  click:function(n){  var addvcd=n.attributes.id;  var text=n.attributes.text;  if(text=='全国'){  addvcd='100000';  }  defaultaddvcd=addvcd;  defaultname=text;  if(analyseflag!=null){  document.getElementById("myiframe").src="analyse.jsp"; }  else{  document.getElementById("myiframe").src=  "prepare.jsp?addvcd="+addvcd+"&title="+encodeURI(encodeURI(text));  }  }  }  });  }); |

**5.3 系统展示**

**5.3.1 系统界面展示**

登录系统后，系统界面如图5.6所示。首页，相似性分析的标识下面是两个选项，单流域搜索和多流域分析，分别点选就能进入各自的相似流域查询功能。在系统右下部分显示数据库中所有的河流信息，左半部分是行政区划表，点选里面的省份或城市，河流信息表就会显示该地的河流信息。河流信息表的上面是新增、更新和上传DEM三个按钮，分别对应新增河流信息、更新河流信息和上传DEM参数文件三个功能。同时还有一个搜索框。

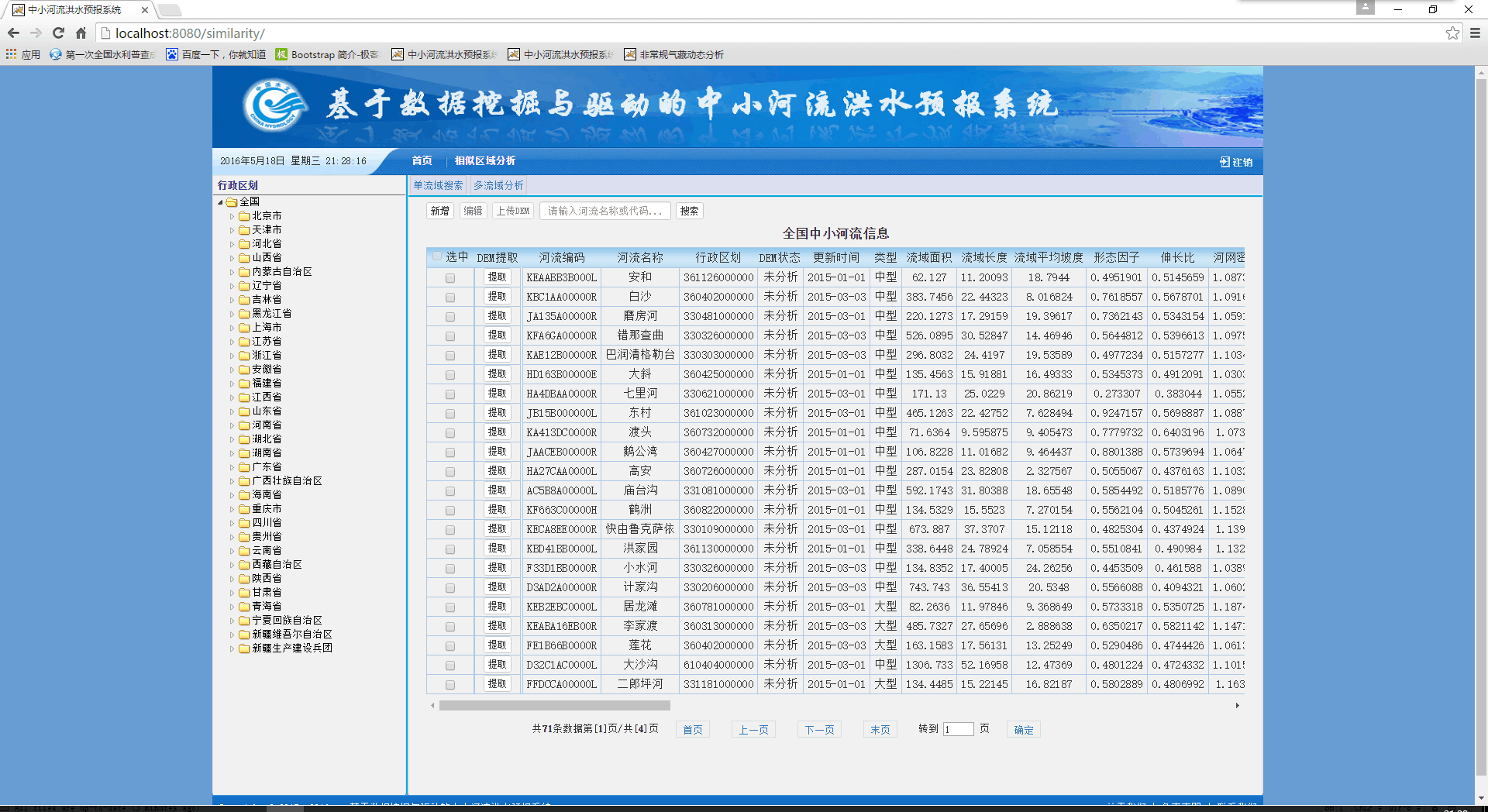


图5.6 系统界面图

**5.3.2 单流域搜索功能**

点击主界面上的单流域搜索功能。在河流信息表中勾选一条作为基准河流的待搜流域，这里以安和为例，勾选安和后，待搜流域下拉框中会出现安和的名字。也可将下拉框下拉，其显示当前页面所有河流的名字，点选其他名字可更换河流。然后勾选河流信息表中其他河流作为搜索范围，可以选不同页的河流，它们的信息会显示到搜索范围的多选框中，在多选框中可以再次确认需要比较的河流。点击重置按钮会清空多选框中的河流；勾选多选框中一部分河流再点击移除时只移除选择的河流。在相似性要素的选栏中勾选需要的分析要素，一般默认全部。最后点击确认，进行分析。如图5.7所示：

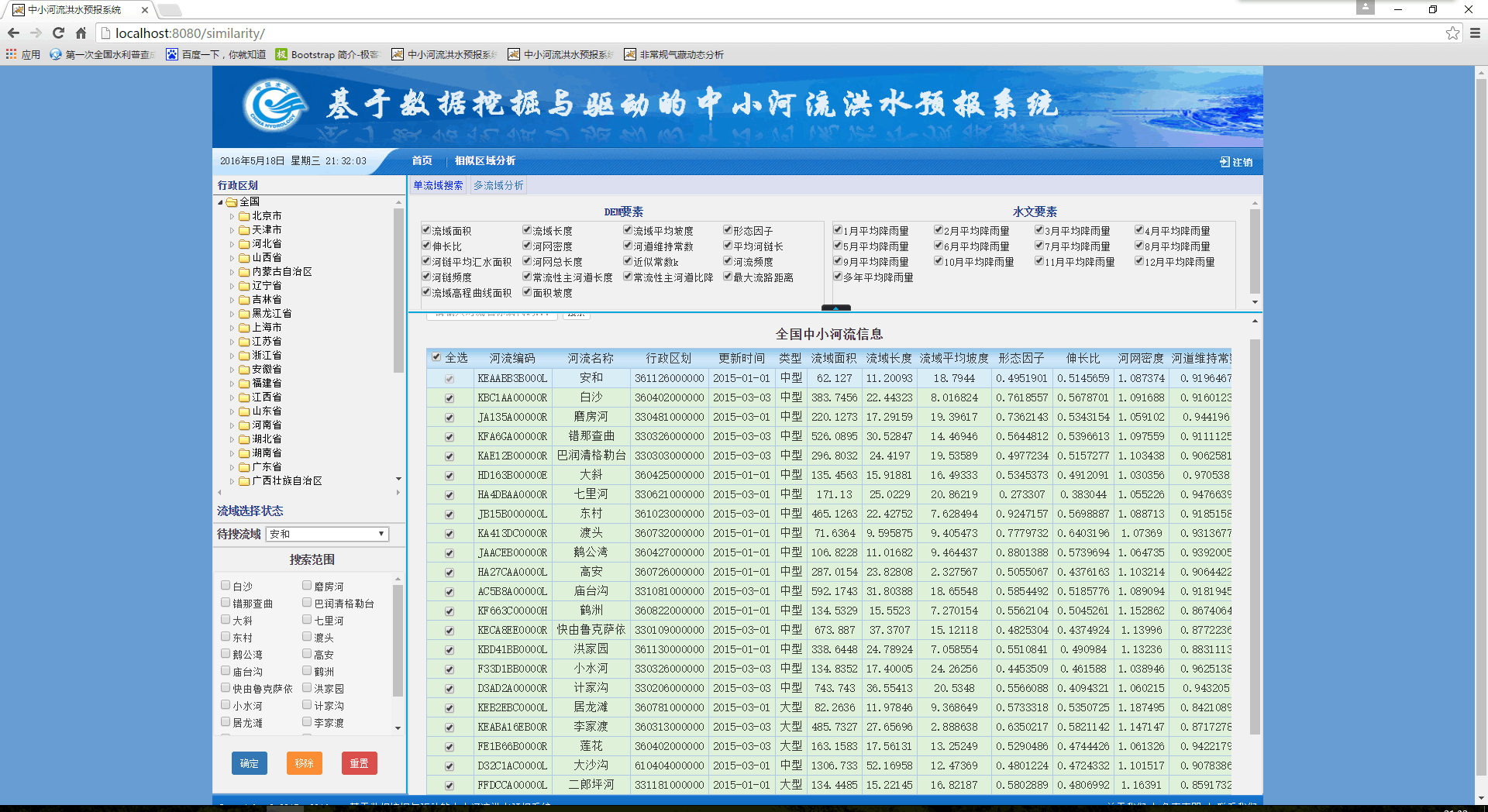


图5.7 单流域搜索

分析完成后结果返回，以聚类图的形式和表格数据的形式进行展示。表格中列出了所有和安和相似的河流，用青色标出。在右上角聚类数目一栏，可以调整用户想要的聚类数目，返回不同结果。如图5.8所示：

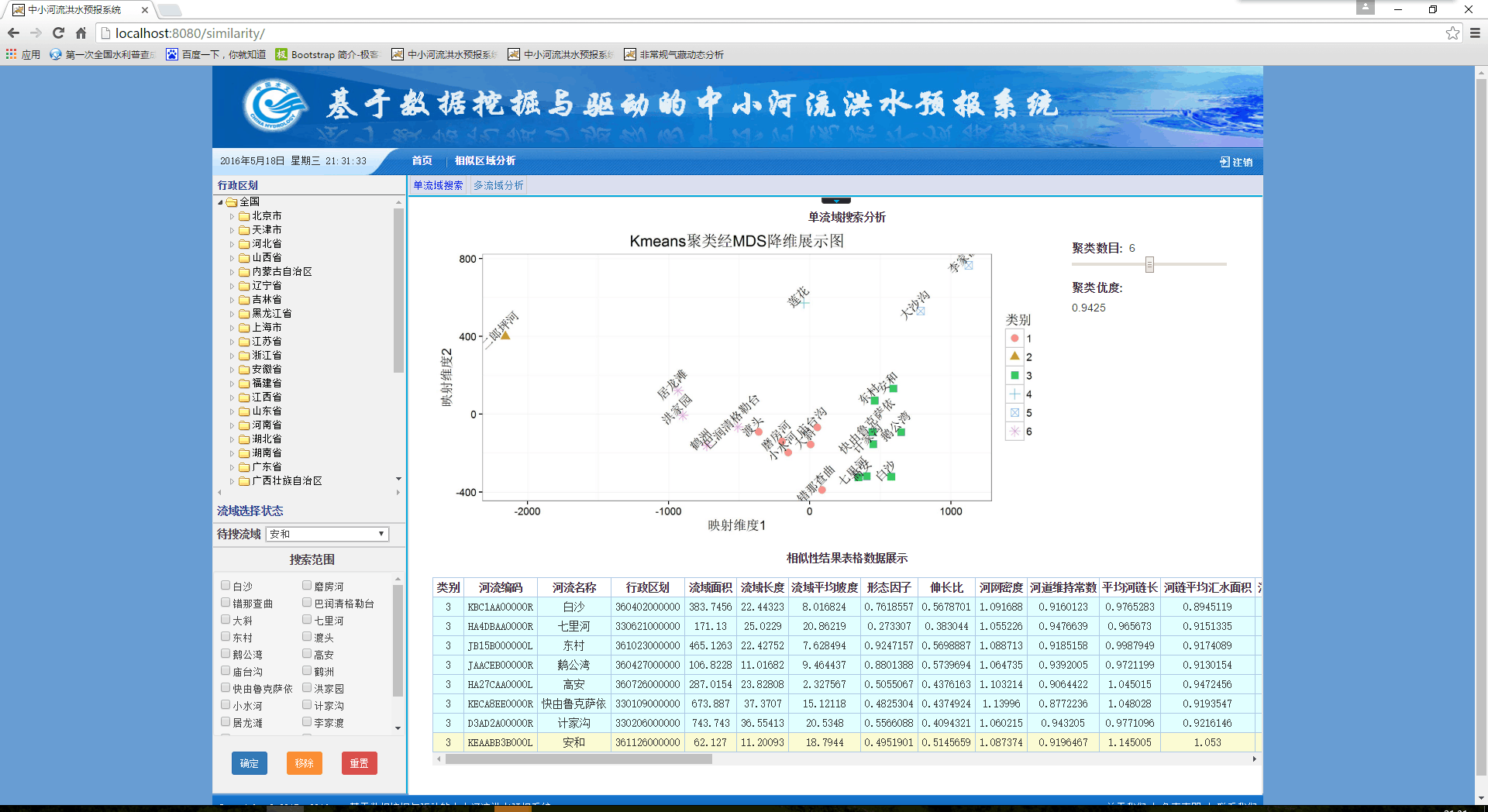


图5.8 单流域搜索结果展示

**5.3.3 多流域分析功能**

与单流域搜索一样在河流信息表中勾选需要的河流，它们会在分析范围中显示。然后在分析范围的多选框中对这些河流进行进一步的筛选操作：勾选不需要的河流，然后点击移除；若全部不需要就点击重置按钮。确认了分析范围后，在相似性要素栏中选择需要的分析要素，一般默认全部要素。最后点击确认进行分析，界面如图5.9所示：

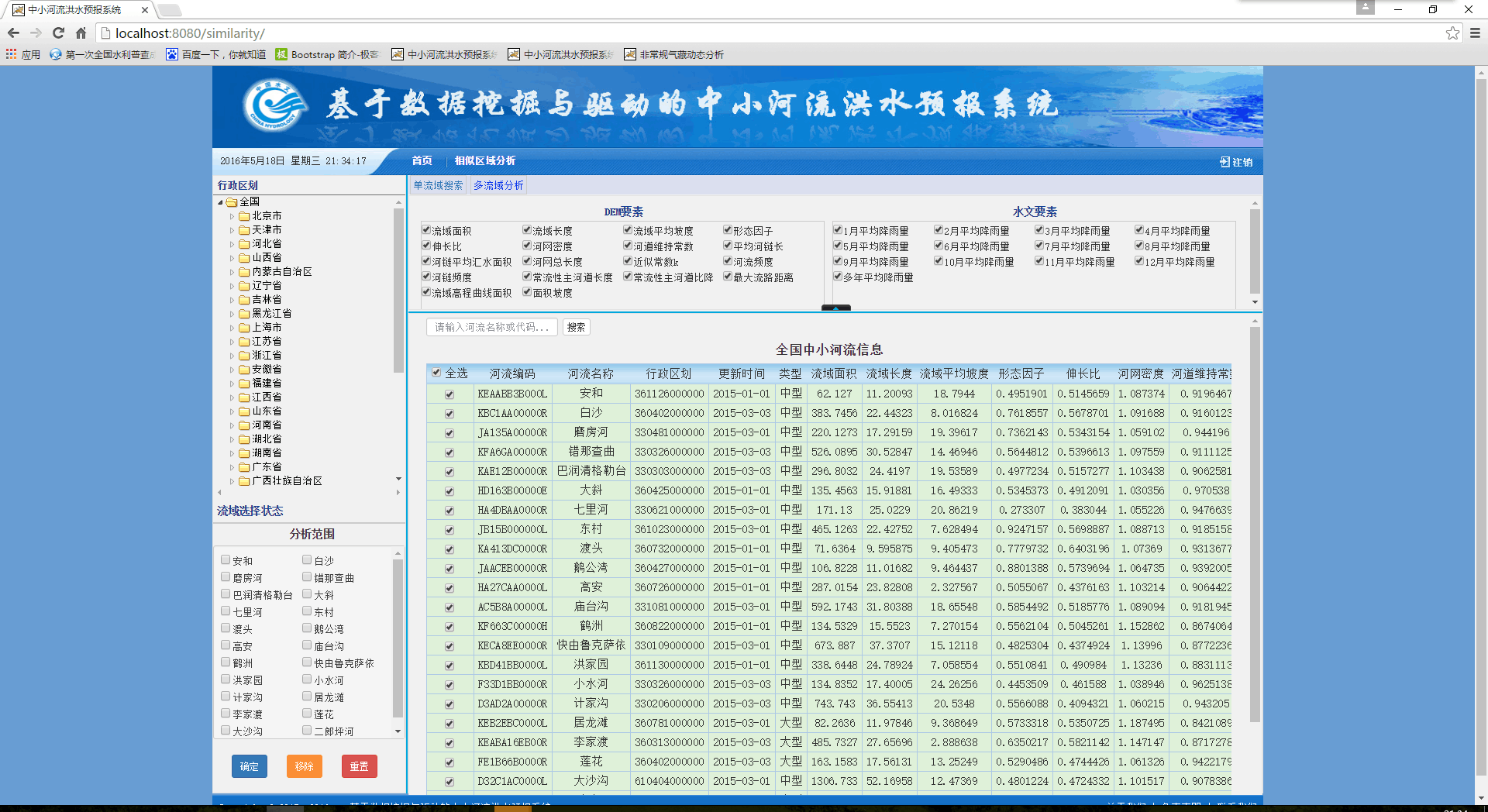


图5.9 多流域分析

分析完成后结果返回，以聚类图的形式和表格数据的形式进行展示，如图5.10所示：

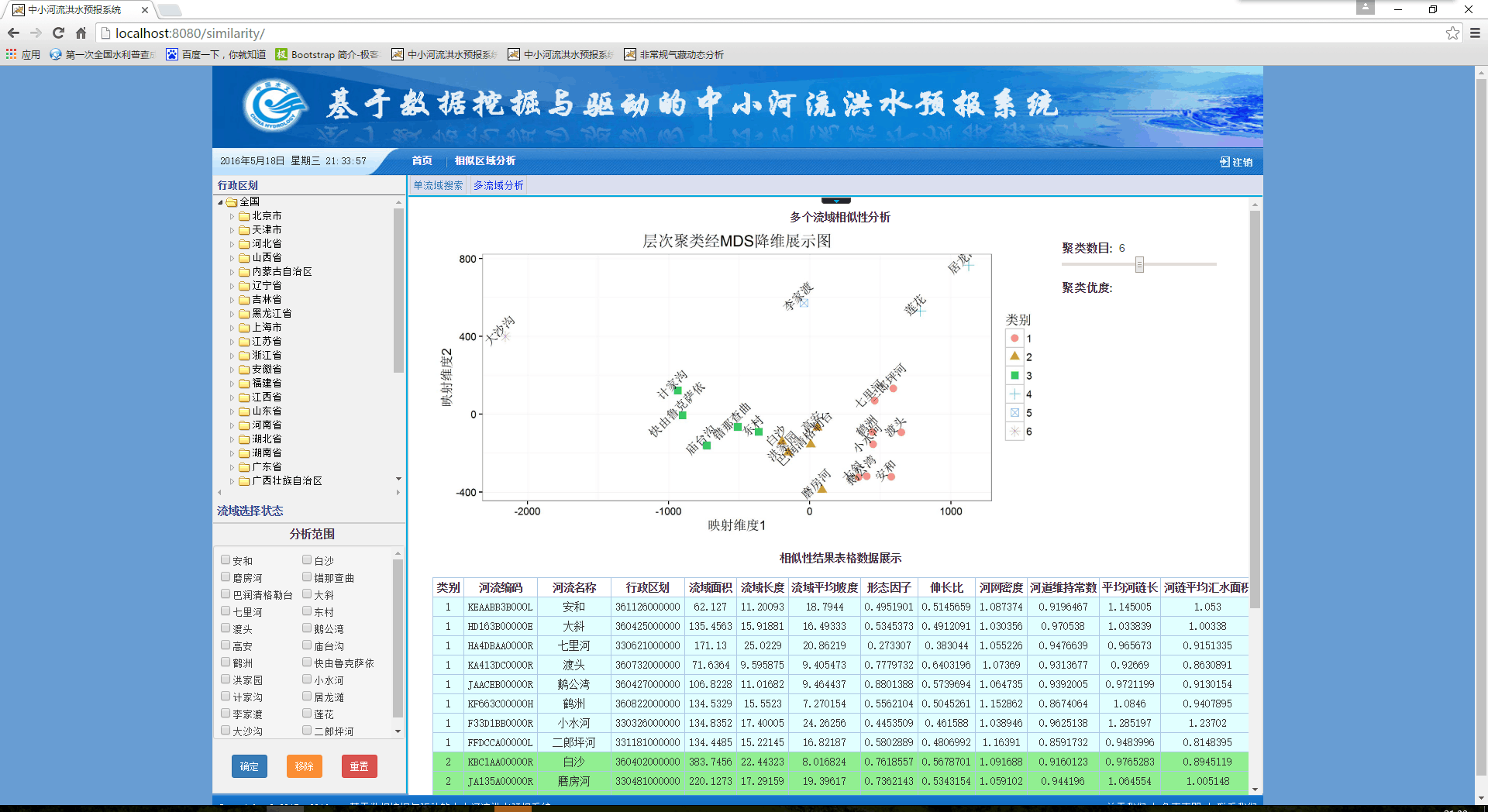


图5.10 多流域分析结果展示

**5.3.4 河流信息查询功能**

在主界面的搜索框中输入河流编码，点击搜索按钮，河流信息表就会显示该河流的信息，如图5.11所示。



图5.11 河流信息查询

**5.3.5 上传DEM参数文件功能**

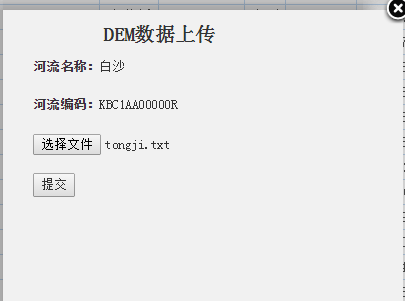
勾选河流信息表中的一条河流，点击上传DEM按钮，会跳出如下界面：

图5.12 上传DEM参数文件

选择上传文件的地址，点击提交，提交成功后会显示“提交成功！DEM状态更新完成。”

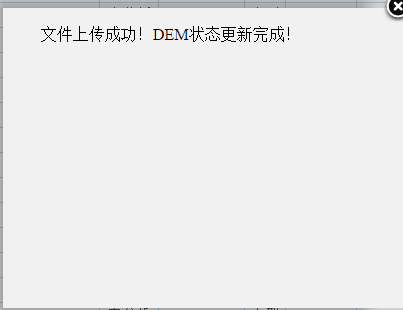


图5.13 上传成功

**5.3.7 新增河流信息功能**

点击新增按钮，在弹出的页面框中填写河流代码、河流名称、行政区划、DEM参数信息和雨量信息，点击提交，其中河流代码、河流名称、行政区划三项不能为空。其界面如下图所示：



图5.14 界面一



图5.15 界面二

**5.3.8 编辑河流信息功能**

勾选一条河流，点击编辑按钮，弹出编辑信息的界面，输入需要修改的值提交即可，页面与新增河流相似。

**5.4 小结**

本章主要介绍了基于DEM的相似流域查找系统的实现过程，并对系统界面进行了展示。

**第6章 总结与展望**

流域相似性研究的展开将加强人们对相似流域的理解，根据流域特点进一步对流域类型进行总结，提出研究模型，完善水文资料，提高流域研究的科学性。随着高分辨率遥感卫星、机载传感系统、全球卫星导航系统、合成孔径雷达等对地观测技术的发展, DEM空间数据步入大数据世代。但原始DEM数据并不能直接表达有效的水文信息，运用数据挖掘技术基于DEM数据进行流域特征挖掘对流域相似性研究有重要意义。

本文将数字高程模型引入流域相似性分析中，设计并实现了基于DEM的流域相似性查找系统。对基于DEM流域特征的提取过程进行了详细描述，以及利用相似性分析中的主成份分析、K-Means算法和层次聚类算法在河流特征信息上进行了实现，并运用web技术对查询结果进行展示。

但是，本系统的DEM特征信息还是基于ArcGIS软件手动提取，如何实现相似性指标自动提取和保证河流相似结果的准确性是以后改进的方向。

**致 谢**

在本次毕业设计过程中，我学习了很多之前没有接触过的技术，并在实践过程中逐渐掌握了这些技术，为我以后的学习打下了坚实的基础。

我首先要感谢我的论文指导老师——万定生教授悉心的指导。在每周的毕设讨论会议上万老师都会询问我的毕设进展情况，并针对毕设中存在的问题，给出解决问题的建议。在论文写作期间，万老师都会对我进行督促。万老师对待工作严谨的态度、渊博的学识、耐心的教诲让我受益匪浅。谨此表示衷心的感谢！

然后还要感谢实验室的沈强、陆宇庆两位师兄，他们在技术上给予我的指导和论文写作上的给我的建议让我的毕业设计进展事半功倍。

最后，请再次让我感谢所有在毕业设计中给予我帮助的人，使得我的毕业设计能顺利如期完成。

参考文献

[1]蔺彬彬. 基于DEM的三川河流域特征提取研究[D]. 太原理工大学,2013.

[2]刘金涛, 宋慧卿, 王爱花. 水文相似概念与理论发展探析[J]. 水科学进展, 2014, 25(2):288-296.

[3]Tribe A. Automated recognition of valley lines and drainage networks from grid digital elevation models: a review and a new method[J]. Journal of Hydrology, 1995, 167(s 1–4):393-396.

[4]Garbrecht J, Martz L W. Digital Elevation Model Issues In Water Resources Modeling[J]. Geocarto International, 2008, 21(1):21-26.

[5]O'Callaghan J F, Mark D M. The Extraction of Drainage Networks From Digital Elevation Data[J]. Computer Vision Graphics & Image Processing, 1984, 27(3):323-344.]

[7]S. K. Jenson and J. O. Domingue, “Extracting Topographic Structure from Digital Elevation Data for Geographic Information System Analysis,” Photogrammetric Engineering and Remote Sensing, Vol. 54, No. 11, 1988, pp. 1593-1600.

[8]Quinn P , Beven K, Chevalier P , et al . The prediction of hillslope flow paths for distributed hydrological modelling using digital terrain models [J ]. Hydrological Processes , 1991 , vol.5 : 59-79.

[9]赵美玲. 基于DEM数据提取水文流域参数分析[J]. 山西水利, 2014(6):46-47.

[10]张振, 宋亚娅. GIS环境下基于 DEM流域水文地理信息的提取[J]. 地下水, 2014(6).

[11]张永红. 基于DEM的辽河流域子流域边界的自动提取[J]. 山东工业技术, 2015(14):262-263.

[12]张国义, 房明惠, 徐云,等. RSI River Tools系统及其应用介绍[J]. 计算机应用, 2002, 22(8):38-40.

[13]Langhaar H L. Dimensional Analysis and Theory of Models[J]. John Wiley & Sons, Inc. New York, N. Y.; Chapman & Hall, Ltd. London, 1951.

[14]RODRIGUEZ-ITRUBE I, VALDES J B. The geomorphologic structure of hydrologic response [J]. Water Resources Research,1979, 15: 1409-1420.

[15]Young A R. Stream flow simulation within UK ungauged catchments using a daily rainfall-runoff model[J]. Journal of Hydrology, 2006, 320(s 1–2):155-172.

[16]丁晶.模糊相似选择在水文计算中的应用[Ｊ].成都科技大学学报,1998,(6).

[17]陈守煜.相似流域选择的模糊集模型与方法[Ｊ].水科学进展,1993,(4).

[18]邓红霞, 李存军, 张少文,等. 基于集对分析的相似流域选择方法[J]. 人民黄河, 2006, 28(7):3-4.

[19]胡余忠, 胡以宝, 章彩霞,等. 基于DEM的流域相似性数值识别与应用[J]. 江淮水利科技, 2010(4).

[20]肖益民, 梅汇海. 相似流域选择的灰关联方法[J]. 武汉水利电力大学学报, 1998(3):10-11.

[21]李亚伟, 陈守煜, 聂相田. 基于PCA和聚类分析的相似流域选择方法[J]. 东北水利水电, 2004, 22(7):1-3.

[22] 赵焱. 基于ArcGIS 10.0的河道地形高程数据处理方法[J]. 电子制作, 2015(5).

[23] <http://wenku.baidu.com/view/8e988c003169a4517723a3f6.html>

附录1：国外经典论文——原文

**A Rough Set-Based Hierarchical Clustering**

**Algorithm for Categorical Data**

Duo Chen, Du-Wu Cui, Chao-Xue Wang, Zhu-Rong Wang

School of Computer Science and Engineering, Xi’an University of Technology,

Xi’an, 710048, China

[vkxtfj@163.net](mailto:vkxtfj@163.net)

**Abstract**

In this paper, rough set theory is applied to the clustering analysis. The clustering decision table is formed through the introduction of decision attribute into data table, thereby further defining the attribute membership matrix. The consistent degree and aggregate degree are present, and their functions in the clustering process are deeply analyzed. The clustering level calculation formula is designed, in which two factors such as consistent degree and aggregate degree are taken into comprehensive account. Also, this paper gives the categorical similarity measure based on Euclidean distance so as to better solve the problem of difficult measurement of categorical data because of the non-numerical data nature. On the basis of the above work, a novel categorical clustering algorithm is designed.

**Keywords:** Clustering; Categorical data; Rough set theory; Similarity measure.

1. **Introduction**

Clustering analysis is an important research project in knowledge discovery and data mining (KDDM). In practical application, the data sets contain numerical and categorical (nominal) data in general. Accordingly, clustering algorithm is required to able to deal with both numerical data and categorical data. K-means algorithm suggested by Mac Queen [1] is one of the most popular clustering algorithms, and it works only on numeric data. Accordingly, K-modes presented by Huang [2] has expanded K-means algorithm so as to deal with categorical data. The fuzzy K-modes algorithm put forward by Huang and Ng [3] has improved the clustering accuracy by using fuzzy processing technology. The above-mentioned algorithms belong to the partition methods in clustering analysis, with high operation efficiency, but there exist the shortcoming that clustering results are often dependent on the selection of the initial points. With an aim at the shortcoming, Bradley and Fayyad [4] posed the refining initial points for k-means clustering; Sun, Zhu and Chen [5] advanced the iterative initial-points refinement algorithm for categorical data clustering; and D. W. Kim, K. H. Lee and D. Lee [6] suggested the fuzzy clustering of categorical data using fuzzy centroids. Concept clustering algorithm [7 ,8, 9] is another kind of method to deal with the clustering problems of categorical data such that this kind of method can not only realize the clustering process, but provide the concept descriptions of clusters as well. The hierarchical clustering algorithm can deal with both numerical data and categorical data. ROCK algorithm [10] is a type of agglomerative hierarchical clustering algorithm for categorical data. Rough set theory (RST) suggested by Pawlak [11, 12] is a new mathematical tool to deal with vagueness and uncertainty, with successful applicable results obtained in many fields of information system. Particularly in recent years, being a kind of favorable mathematical tool, RST has displayed the vast applicable future in KDDM field. It can be expected that RST holds a vast applicable promise either in theoretical research field or in the practical application.

In dealing with the categorical data, one of the difficulties is to resolve the problem of similarity measure, for the nature of categorical data is the non-numerical so that Euclidean distance extensively-used in numerical data processing can’t be employed directly. However, in this paper, RST is applied to clustering analysis; and the clustering data set is mapped as the decision table through introducing a decision attribute, whereby configuring the attribute membership matrix and presenting clustering consistent degree and aggregate degree and analyzing their functions in the clustering process. Also, the categorical similarity measure based on Euclidean distance is suggested. This measure is better to solve the problem of difficult measurement because of non-numerical nature of categorical data. Based on the analysis, this paper designs the rough set-based hierarchical clustering algorithm for categorical data. Theoretical analysis and experimental results indicate that this algorithm is valid.

1. **Basic RST Notions**

This section briefs on the basic notions of RST used in this paper and the detailed definitions can be referred to some related literatures [11, 12 ,13].

**Definition 2.1** An information system (IS, sometimes called data table, attribute-value system, etc.) is a pair (U,A), where U is a nonempty, finite set of objects called the universe and A is a nonempty, finite set of attributes, such that a: U→Va for any a∈A, where Va is called the domain of attribute a.

Each nonempty subset B ⊆ A determines an indiscernibility relation.



RB partitions U into equivalence classes.



Where [x] B denotes the equivalence class determined by x with respect to (wrt) B,

i.e.,

A decision table (DT) is an IS (U, A U D), where A∩D=Ø. Then term A is called the condition attribute set, and D is called the decision attribute set. If R A ⊆ R D , then (U, A U D) is consistent, otherwise it is inconsistent. In general D has only one attribute d.

**Definition 2.2** Let (U, A U {d}) be a DT, B ⊆ A, and U/R {d} ={D1 ,...,Dr}. A

membership distribution function µB:U→[0,1]r is defined as follows:



Where



1. **Rough Set-Based Clustering**
   1. **Related Definitions and Theorems**

In this paper clustering problems are described using the clustering DT

(U, A U{d}), in which U is the universe, one element xi∈U is called as an object, A is the attribute set, A={a1 , … ,as}, and d is the introduced decision attribute with the domain Vd∈{1,...r}, r≤|U|. The main purpose of introducing the decision attribute d is to use it to express the clusters. In terms of RST, the attribute d determines an indiscernibility relation R{d} , which partitions U into equivalence classes, U/R{d}={D1 ,…,Dr}=P, named as clustering model in this paper. It can be considered that the clustering model P expresses a kind of clustering result.

**Definition 3.1** Let (U, AU{d}) be a DT, and P=U/R{d}={D1 , ... ,Dr }, A={a1 ,..., as },n=|U|, an attribute membership matrix Mk is defined as follows:



Obviously, Mk=[µk(i,j)] is n×r matrix and µk(i,j)∈[0,1].Definition 3.1 comes from Definition 2.2, which can be considered as a special example when the condition

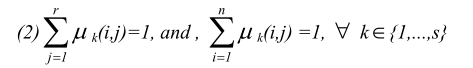
attribute subset B ⊆ A in Definition 2.2 takes the individual condition attribute ak∈A.It is just because the individual conditional attribute is taken that the membership

matrix Mk has the different features from those of membership distribution function as well as different manipulating methods.

**Theorem 3.1** Let (U, A U{d}) be a DT, and A={a1 ,..., as }, n=|U|, if P=U/Rd

={D1 ,...,Dn}={{x1 },...,{xn}}, and then,

1. Mk is n-order symmetric matrix, i.e., µk(i,j)=µk (j,i), ∀k∈{1,...,s}



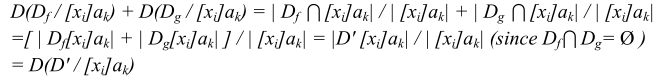
Proof: It follows immediately from Definition 3.1.

**Theorem3.2** (Mergence Theorem) Let (U, A U {d}) be a DT, and A={a1 ,..., as },

P=U/R{d} ={D1 ,...,D r }, n=|U|. If ∀ f, g∈{1,...r}, f≠g and D'=DfU Dg , and then



Proof. ∀ k∈{1,...,s} and i∈ {1,...,n}, we have:



In terms of Mergence Theorem, the merging operation in clustering process can be achieved via the corresponding addition operation in attribute membership matrix, being suitable to either the cluster or the individual object. Therefore, Mergence

Theorem is of great importance for clustering algorithm.

**Deduction 3.2** Let (U, A U {d}) be a DT, and P=U/R{d} ={D1 ,...,D r },A={a1 ,..., as }, n=|U|. If f, g∈{1,...r}, f≠g and D'=DfU Dg,then ∀ i∈{1,...,n}, k∈ {1,...,s}, we have:





where k' is the sequence number of the cluster D' .

* 1. **Consistent Degree and Aggregate Degree**

In the case of traditional RST, the consistent DT and inconsistent DT are only given, and the consistent measure of the DT is not defined. In fact, in a clustering DT (U,A U {d}), for each ak∈A, the jth column of Mk represents the membership distribution of each object in U corresponding to the cluster Dj in clustering model P. This paper holds that this distribution can include the information of coordination of the condition attribute set A to the clustering model P in the DT, further reflecting that the clustering accuracy can be used as a kind of measure in clustering process. On the basis of the above analysis, we have offered the consistent degree definition.

**Definition 3.2** Named mapping Φ:[0,1]→[0,1] as the coordination, if satisfying the following properties:

(1) Φ (x) = 1 iff x= 0, 1, and Φ (1/2)=0

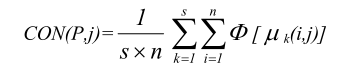
(2) Φ (x) = Φ (1-x)

(3) Φ (x) on [0, 1/2] monotonic reduction

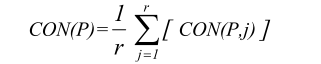
In Definition 3.2, condition (1) indicates that when independent variable x is 1 or 0, i.e. the object completely or incompletely falls under the specified cluster, the coordination can reach its maximum value 1; correspondingly, when x is 1/2, i.e. the membership relation is entirely unclear such that the coordination reaches its minimum value 0; condition (2) indicates that the coordination is about x=0.5 symmetry; condition (3) specifies monotonic features of the mapping.

We introduce Φ(x)=2×|x-0.5| served as the coordination function.

**Definition 3.3** Let (U, A U {d}) be a DT, P=U/R{d}={D1 ,...,D r }. We define the consistent degree of the Dj in P as:

(1)

and the consistent degree of the DT as:

 (2)

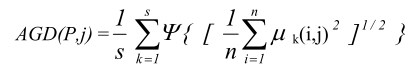
Obviously, CON(P, j), CON(P)∈[0,1].

**Theorem 3.3** In (U, A U{d}), if P=U/R{d}={D1 ,...,D r }={U}, then CON(P)=1.

Proof. It follows immediately form the definition of CON(P)

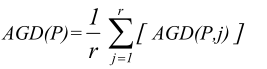
Without considering the case of P= {U}, the larger the CON(P) is, the higher the clustering accuracy is. The clustering process should render CON(P) to advance in the direction of enlargement. But it is not complete to guide the clustering via CON(P), this is because consistent degree does not include the clusters in P upon the objects containing information. This paper ushers the aggregate degree to indicate that the clusters contain the objects to a certain extent.

**Definition 3.4** Let (U, A U {d}) be a DT, n=|U| ,A={a1 ,..., as }, P=U/R{d}={D1 ,...,D r }. we define the aggregate degree of Dj in P as:

（3）



and the aggregate degree of the DT as:

（4）

In Definition 3.4, the mean-root-square is employed to define AGD(P, j), whereas the arithmetic mean value is not adopted to do the definition. This is mainly due to avoiding yielding trivial values (i.e. simple 0 or 1). This method borrows the predictiveness definition method listed in literature [9]. But the predictiveness and aggregate degree have three points in difference: the first is that the predictiveness is defined in terms of probability distribution, while the aggregate degree is defined in terms of membership matrix; the second is that the aggregate degree has had the normalization manipulation, whereby avoiding that the aggregate degree can quickly reach rather large numerical values with an increase in cluster numbers; and the third is that ushering in Ψ function is convenient to match with consistent degree. As far as Ψ function is concerned, this section will analyze it in details in later part.

**Theorem 3.4** Let (U, AU{d}) be a DT, P=U/R{d}={D1 ,...,D r },then AGD(P)∈(0,1].

Proof. For any ak∈A, since µk(i,j)∈[0,1], we have AGD(P)≤1; Because if elements in any column of Mk are all zero, indicating that any objects in U do not fall under the cluster corresponding to this column in P, this is impossible, so we have AGD(P)>0. To sum up the above descriptions, AGD(P)∈(0,1] holds.

**Theorem 3.5** Let (U, AU{d}) be a DT, P=U/R{d}={D1 ,...,D r }, and D'=Df U Dh ,

∀ f, h∈{1,...r} , f≠h. If P'=(P- { Df ,Dh })U{D'}, then AGD(P') > AGD(P).

Proof. It follows immediately form Definition 3.4 and Merge Theorem.

In terms of Definition 3.4 and Theorem 3.5, AGD(P) expresses the containing degree of clusters in P upon the objects. In agglomerative hierarchical clustering algorithm, from the starting P including n clusters, i.e. each object being a cluster to the final P containing one cluster, i.e. all the objects being a cluster, has formed the dynamic clustering map. In the dynamic clustering process, aggregate degree goes up monotonically but the consistent degree maybe not. Actually in practical application, the consistent degree has some reductions with an increase in aggregate degree. The nature of monotonic increase of aggregate degree is of important application value. We may appoint the threshold λ∈(0,1) of the aggregate degree as the algorithm ending condition. An effective algorithm should take the two indexes of CON(P) and AGD(P) into comprehensive consideration, whereby rendering CON(P) to have some increase or to have minimal decrease with an increase in AGD(P) in the clustering process, for this reason the calculation formula of clustering level is introduced.

**Definition 3.5** Let (U, AU{d}) be a DT, P=U/R{d}={D1 ,...,D r }. we define the clustering level of Dj in P as:

（5）

And the clustering level of the DT as:

（6）

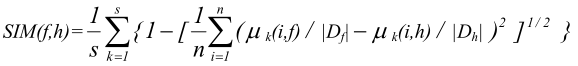
Clustering level can comprehensively reflect two indexes of the consistent degree and the aggregate degree. It is easy to proof that LEV(P) is the monotonic increasing function wrt AGD(P) and CON(P). Obviously, LEV(P) ∈ [0,1], and LEV(P)=0, iff CON(P)=0; LEV(P)=1, iff CON(P)=AGD(P)=1. It can be known from Definition 3.5 that AGD(P) and CON(P) in LEV(P) are two symmetrical parameters. When AGD(P) and CON(P) values are closer, they will have the similar effect upon LEV(P); when the value of one of the parameters is much smaller than that of other parameter, the parameter having the small value will have the greater effect upon LEV(P).

The function Ψ is introduced into AGD(P) definition (Definition 3.4). This is because in the initial stage of agglomerative algorithm, AGD(P) values are small (or very small when the data set are very large) so that the main purpose of introducing function Ψ lies in raising AGD(P) value in the initial stage of algorithm so as to match with CON(P). Without introducing function Ψ and in the initial stage of algorithm, AGD(P) in the clustering level is too small to render CON(P) to play its role, which is no doubt to affect the accuracy of clustering algorithm.

* 1. **The Categorical Similarity Measure Based on Euclidean Distance**

In this section, a novel categorical similarity measure is suggested.

**Definition 3.6** In (U, A U{d}),A={a1 ,..., as }, P=U/R{d}={D1 ,...,D r }, n=|U|, ∀ f, h∈{1,...,r}, , the similarity between two clusters Df and Dh in P can be defined as follows:

 （7）

Definition 3.6 cites Euclidean distance to describe the similarity among clusters such that the greater the value of distance is, the smaller the value of similarity is. Not only can the similarity defined using Euclidean distance measure the numbers of the same attributes between two clusters and differences in the numbers of dissimilar attributes, but also the key lies in expressing the degrees of the similar and dissimilar attributes, whose nature is to do numerical processing of categorical attributes. In Eq.(7), µk(i,f)/|Df| and µk(i,h)/|Dh| indicate the centers of cluster Df and cluster Dh respectively, i.e. using membership mean value represents the cluster centers in such a way as to borrow the expressing method of the well-known K-means algorithm[1] in numerical clustering. If Df and Dh contain one object, then |Df|=| Dh |=1. Accordingly Definition 3.6 is also adaptable to this situation so that Definition 3.6 expresses the similarities among the cluster versus the cluster, and the object versus the cluster as well as the object versus the object.

* 1. **The Algorithm**

This section presents the Rough Set-Based Agglomeration Hierarchy Clustering Algorithm (RAHCA). This algorithm first lets P=U/R{d}={{x1},...,{xn}}, and then conducts merging operation in terms of the clustering level and similarity measure. This operation can be carried out till the number of clusters m or the aggregate degree threshold λ given by users and outputs the clustering results, i.e. clustering model P. This algorithm can also be conducted till all the objects merged into one cluster. In such a way the algorithm outputs the dynamic clustering map.

**Algorithm RAHCA**

Input: Data table (U,A), number of clusters m (or aggregate degree threshold λ)

Output: Clustering model P

Step 1 Let P(0)= {D1 ,...,Dr }={{x1},...,{xn}}, n=|U|, r=n.

Step 2. In terms of Definition 3.1, derive the membership matrix Mk, k ∈ {1,...,|A|}

Step 3. Repeat the following operations, until r=m (or LEV(P)≥λ)

Step 3.1 Find the cluster Dmin with the minimum LEV value using Eq.(5).

Step 3.2 Compute the similarity between Dmin and the rest of the clusters in P using Eq.(7), let Dsim be the cluster scoring the highest similarity with Dmin .

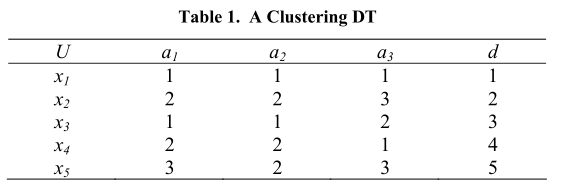
Step 3.3 Update clustering model P and membership matrix Mk , k∈{1,...,|A|} by merging Dmin and Dsim according to Mergence Theorem, and then r =r-1.

The algorithm complexity is analyzed as follows. In step 2, the equivalence classes

U/R{ak} should be first computed for computation Mk , and then r×|U| elements in Mk must be conducted, where r is the number of the clusters in P with the maximum |U|. Algorithm 1 in literature[14] can be used to compute equivalence classes, whose time complexity is O(|A||U|)log(|U|), thereby the time complexity of step 2 should be O(|A||U|log|U|)+O(r|A||U|). In step 3, the maximal number of the iterations is |U|-1,and each iteration needs to compute LEV r times in step3.1 and SIM r-1 times in step3.2, both of them can be computed in O(r|A||U|), and the updating operations in step 3.3 can be finished in O(|A||U|), such that the time complexity of step 3 is O(r|A||U|2). So the time complexity of the algorithm can be estimated as O(|A||U|3) . We can easily analyze that the space complexity is O(r|A||U|2). It is worth mentioning that the above-mentioned analysis is in the most unfavorable situation and regards Mk as the dense matrix, but in practical application M k is the highly sparse matrix. Using a type of appropriate sparse matrix operating strategy, both the time complexity and the space complexity of the algorithm will be reduced notably.

1. **Experimental Results**
   1. **Artificial Data Set**

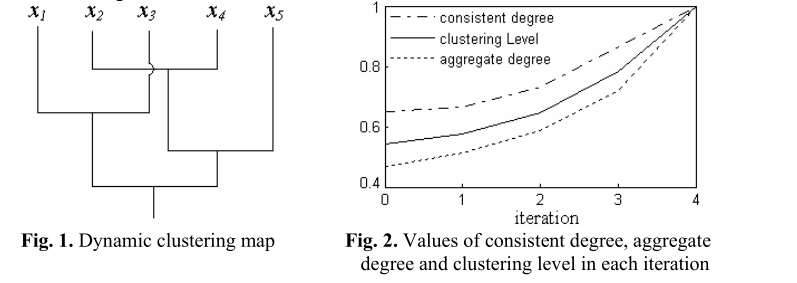
In this section, one simple example is used to demonstrate the execution of RAHCA. In the clustering DT shown in Table 1, the universe U={x1,...x5}, the initial categorical condition attribute set A={a1, a2 ,a3}, d is the introduced decision attribute with the domain Vd={1,...,5}, and it is in its initial state as shown in Table 1.



In the first iteration, the LEV values of Dj , j∈{1,...,5} are 0.523, 0.515, 0.591,

0.515 and 0.581 respectively, LEV[P(0),2]=0.52 is the minimum, let Dmin=D2 , the SIM values between D2 and the rest objects are 0.57, 0.53, 0.85 and 0.82 respectively, obviously the nearest cluster is D4 , SIM(2,4)=0.85. By merging D2 and D4, P(1)={x1,{x2 ,x4},x3,x5} can be obtained. In terms of the same computing, in the second iteration by merging D1 and D3, P(2)={{x1 ,x3},{x2 ,x4},x5} can be obtained. In the third iteration, P(3)={{x1 ,x3},{x2 ,x4,x5}}; and in the last iteration, P(4)={{x1,x2,x3,x4 ,x5}}, thus the algorithm comes to its end. It is easy to analyze that P(2) and P(3) are the rational results.

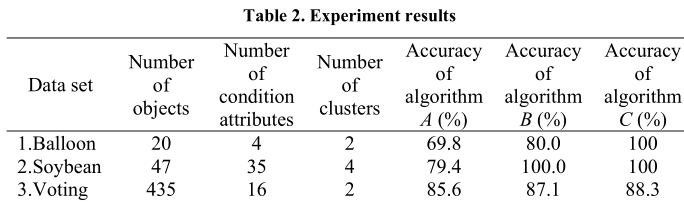
The dynamic clustering map is shown in Fig.1, and the curves that indicate the changing in numerical values of CON(P), AGD(P) and LEV(P) in each iteration are shown in Fig. 2.



* 1. **UCI Data Sets**

We ran experiments on 3 data sets obtained from the UCI Machine Learning Repository. The Balloon date set contains 20 instances and each instance has 4 categorical attributes. It is classified into 2 classes. The Soybean date set contains 47 instances on diseases in soybeans and each instance has 35 categorical attributes. The data set is classified into 4 classes according to its disease type. The Voting data set derives from 1984 United States Congressional Voting Records Database. It contains 435 instances, which represent the voting records of 267 democrats and 168 republicans respectively. Each instance has 26 Boolean attributes. There are some missing attribute values in the date set and they are regarded as a special constant value in this paper.

The algorithm in literatures [2, 9] and RAHCA algorithm proposed in this paper are adopted respectively (algorithm A, algorithm B and algorithm C for their short forms). The experiment results are shown in Table 2. In experiments the standard numbers of clusters are specified. The accuracy of an algorithm is the ratio of the total number of instances occurring in both the ith cluster and its corresponding true class to the number of instances in the data set. The results of Algorithm A are the average accuracy value of 50 runs. The experiment results indicate that the Algorithm C, i.e. RACHCA, has the highest accuracy for all of above date sets.



1. **Conclusions**

This paper applies RST to the clustering analysis in KDDM, and introduces the decision attribute to configure the clustering DT, whereby defining the membership matrix. This paper suggests the consistent degree and aggregate degree measures corresponding to the clustering model P of the DT. The two measures express two aspects of clustering process respectively. The consistent degree expresses the coordination degrees among the equivalence classes of the condition attribute set and the clusters in clustering model P of the DT, whereas aggregate degree indicates the containing degree of cluster itself upon the objects. In order to take the effect of the two factors upon the clustering process into comprehensive consideration, the clustering level calculation formula is designed, in which the consistent degree and aggregate degree are in the symmetric positions, whereby rendering the clustering process to be able to give consideration to the two measures at the same time. When their values are near, they can affect the clustering level in common; when one of which is smaller, this measure will play an important role in the clustering level. It is just for this reason that it can become a major factor to affect the clustering direction. In practical application, the aggregate degree is small in the initial clustering stage, so the aggregate degree is a major influencing factor in algorithm, and it is just at this time that the aggregate degree in algorithm should be raised. With the clustering ongoing, the aggregate degree increases gradually, and it, together with the consistent degree, guides the clustering. By the late stage of clustering, the consistent degree becomes a major influencing factor because of its becoming smaller such that it is just at this time that algorithm should first guarantee the consistent degree.

Also, this paper poses the definition of similarity measure for categorical data based on Euclidean distance, whereby the problem of difficult comparison of similarity caused by the nature of non-numerical values in categorical data can be better resolved. This measure is adaptable to the comparison among the cluster versus the cluster and the cluster versus the object as well as the object versus object. In addition, the consistent degree, aggregate degree and similarity measure are normalized in such a manner as to be convenient for the manipulation of each parameter and to make these parameters more coordinated and rationally brought into full play.

Based on the above analytical results, this paper designs the Rough set-based Agglomerative Hierarchical Clustering Algorithm (RAHCA) suitable for categorical data. This algorithm needs the users to offer the number of clusters or the aggregate degree threshold. Without these parameters specified, the algorithm outputs the dynamic clustering map which needs to be further analyzed so as to obtain the clustering results. The future research work will include the further improvement of algorithm efficiency as well as research on clustering algorithm for mixed numeric and categorical data.

**References**

[1] Mac Queen, J, "Some methods for classification and analysis of multivariate observations", Proceedings of the 5th Berkeley Symposium on Mathematics, Statistics and Probability, 1967, pp. 281-297

[2] Huang, Z., "Extensions to the K-means Algorithm for Clustering Large Data Sets with Categorical Values", Data Mining and Knowledge Discovery Ⅱ. 1998, pp. 283-304

[3] Huang, Z., Ng, M.K., "A Fuzzy K-Modes Algorithm for Clustering Categorical Data". IEEE Trans. Fuzzy Systems, vol.7, 1999, pp.446-452

[4] Bradley, P., Fayyad, U., "Refining initial points for k-means clustering", Proceedings of the 15th International Conference on Machine Learning. Morgan Kaufmann, Los Altos, CA.1998

[5] Sun.Y., Zhu, Q.M., Chen, Zh. X., "An iterative initial-points refinement algorithm for categorical data clustering", Pattern Recognition Letters, vol. 23, 2002, pp. 875-884

[6] Kim, D.W., Lee, K. H., Lee. D., "Fuzzy clustering of categorical data using fuzzy centroids". Pattern Recognition Letters, vol. 25, 2004, pp. 1263-1271

[7] Michalski R., Stepp, R., "Automated Construction of Classifications: Conceptual Clustering Versus Numerical Taxonomy", IEEE Trans. Pattern Anal. Machine Intell., vol. 5, 1983, pp.369-410

[8] Fisher, D., "Knowledge Acquisition via Incremental Conceptual Clustering". Machine Learning, vol. vol. 2, 1987, pp. 139-172

[9] Luis Talavera and Javier Béjar, "Generality-Based Conceptual Clustering with Probabilistic Concepts", IEEE Trans. Pattern Anal. Machine Intell., vol. 23, 2001, pp. 196-203

[10] Guha, S., Rastogi, R., Shim, K., Rock: "A Robust Clustering Algorithm for Categorical Attributes", Proc. Int. Conf. Data Engineering (ICDE'99), Sydney, Australia, 1999, pp. 512-521

[11] Pawlak, Z., "Rough sets", International Journal of Computer and Information Science, vol.11, 1982, pp. 341-356

[12] Pawlak, Z,: "Rough Sets: Theoretical aspects of reasoning about data", Kluwer Academic Publishers, London , 1991

[13] Zhang, W.X., Mi, J. Sh., Wu, W. Zh., "Approaches to Knowledge Reductions in Inconsistent Systems", International Journal of Intelligent System, vol. 18, 2003, pp. 989-1000

[14] Liu Sh. H. et al: "Research on efficient algorithms for rough set methods", Chinese Journal of Computers, Vol. 26, 2003.5, pp.524-529(in Chinese)

附录2：国外经典论文——翻译

**一种基于粗糙集的分类数据层次聚类算法**

Duo Chen, Du-Wu Cui, Chao-Xue Wang, Zhu-Rong Wang

School of Computer Science and Engineering, Xi’an University of Technology,

Xi’an, 710048, China

vkxtfj@163.net

**摘要**

本文将粗糙集理论应用于聚类分析中。聚类决策表通过将决策属性引入到数据表中构成，从而进一步定义属性成员矩阵。符合度和聚合度的存在，并深入分析其在聚类过程中功能。聚类水平计算公式被设计成全面考虑2个或多个因素，比如说符合度和聚合度这两个因素。另外，本文给出了基于欧几里德距离的分类相似性度量，以便更好地解决非数值数据性质分类数据的困难测量的问题。在上述工作的基础上，设计了一个新颖的分类聚类算法。

**关键词：**聚类，分类数据，粗糙集理论，相似度量

1. **简介**

聚类分析是知识发现和数据挖掘（KDDM）的重要研究项目。在实际应用中，数据集包含数值和分类（名义）数据。因此，聚类算法需要能够同时处理数值数据和分类数据。由Mac Queen提出的K-means算法[1]是最流行的聚类算法之一，它仅适用于数字数据。因此，黄[2]提出的K-模式扩展的K-means算法，以便处理分类数据。黄和吴[3]提出的模糊K-模式算法通过使用模糊处理技术提高了聚类的精度。上述算法属于聚类分析中的分区方法，具有较高的运行效率，但存在聚类结果往往由初始点的选择决定的缺点。针对这一缺陷，布拉德利和法耶德[4]提出精炼初始点的k-means聚类；孙，朱，陈[5]提出对分类数据聚类迭代初始点细化算法；D.W.基姆·H.Lee和李政道[6]建议采用模糊质心分类数据模糊聚类。概念聚类算法[7,8,9]是另一种处理分类数据聚类问题的方法，这种方法不仅能够实现聚类过程，还能提供集群的概念描述。层次聚类算法既可以处理数值数据也可以处理分类数据。Rock算法[10]是一种处理分类数据的凝聚层次聚类算法。Pawlak[11,12]提出的粗糙集理论（RST）是一种新的处理模糊性和不确定性的数学工具，在信息系统中的许多领域获得成功应用。特别是近年来，作为一种良好的数学工具，RST已经在数据挖掘中显示出广阔的应用前景。可以预见RST不论是在理论研究领域还是在实践应用中都有良好的前景。

处理分类数据的难点之一是解决相似度量的问题，分类数据的性质是非数值，在数值数据处理中广泛使用的欧氏距离不能直接采用。然而，在本文中RST被应用于聚类分析；聚类数据集映射为决策表，通过引入决策属性，即配置属性成员表和呈现聚类一致度和聚合度并分析其在聚类过程中的功能。同时，提出了基于欧氏距离的分类相似性度量。这一措施更好的解决了由于分类数据的非数值性质导致的难以测量的问题。基于所述分析，本文设计了一种基于粗糙集的分类数据分层聚类算法。理论分析和实验结果表明，该算法是有效的。

1. **RST的基本概念**

本节介绍RST中用到的基本概念和详细的定义可以参考有关文献[11,12,13]。

**定义2.1** 信息系统（IS，有时也被称为数据表，属性值系统等）是一对（U,A），其中U是一个非空有限对象集合和A是一个非空的对象的属性集合，像a:U→Va,,对任意a∈A，Va是属性a的域。

每个非空子集B⊆A确定一个不可分辨关系。



RB将U分割成等价类。



其中，[X] B表示由X相对于（WRT）B决定的等价类，即:



决策表（DT）是一个信息系统(U,A U D)，且A∩D=Ø。然后A称为条件属性集，D称为决策属性集。若RA ⊆ RD,那么(U,A U D)是一致的，否则是不一致的。一般来说，D只有一个属性d。

**定义2.2** 当(U,A U{d})是一个决策表,B⊆A，且U/R{d}={D1 ,...,Dr}。一个成员分布函数µB:U→[0,1]r为如下定义：



当 

1. **粗糙集聚类**

**3.1相关的定义和定理**

在本文中，聚类问题用聚类决策表DT(U,A U{d})描述，其中U是域，一个元素xi∈U被称为一个对象，A是属性集，A={a1 , … ,as}并且D是引入决策属性与域Vd∈{1,...r}, r≤|U|。引入决策属性d的主要目的是用它来表达集群。就RST而言，属性d决定了不可分辨关系R{d}，R{d}将U划分成等价类，U/R{d}={D1 ,…,Dr}=P，本文命名其为一种聚类模型。可以认为，聚类模型P表示一种聚类结果。

**定义3.1** 让(U,A U{d})成为一个决策表，并且P=U/R{d}={D1,...,Dr }, A={a1 ,..., as },n=|U|，一个属性成员矩阵Mk可定义如下：



显然，Mk=[µk(i,j)]是一个n×r矩阵且µk(i,j)∈[0,1]。定义3.1来源于定义2.2，当它在定义2.2中的条件属性子集B⊆A取用个体条件属性ak∈A时可以被视为特例。这只是因为个体条件属性是采用成员矩阵Mk，它具有与其他成员分布函数不同的操作方法与特征。

**定理3.1** 让(U, A U{d})成为一个决策表,并且A={a1 ,..., as }，n=|U|, 若P=U/Rd={D1 ,...,Dn}={{x1 },...,{xn}}，那么：

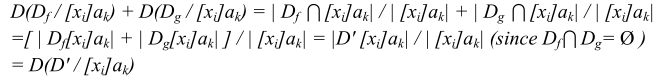
1. Mk是n阶对称矩阵，µk(i,j)=µk(j,i)，∀k∈{1,...,s}。
2. µk(i,j)行和与列和都为1，∀k∈{1,...,s}。

证明：它紧随定义3.1。

**定理3.2**（定理融合）决策表(U,A U{d})且A={a1 ,..., as },P=U/R{d} ={D1 ,...,D r }，n=|U|。若∀f，g∈{1,...r}，f≠g且D'=DfU Dg，那么：



证明：任意k∈1,...,s}和i∈{1,...,n},存在，



根据合并定理，聚类过程中的合并操作可以通过在属性成员矩阵中进行相应的加法运算算来实现，集群或单个对象都适合。因此，合并定理是对聚类算法具有重要意义。

**推论3.2**决策表(U,A U{d})且P=U/R{d}={D1 ,...,D r }，A={a1 ,..., as }，n=|U|。若f，g∈{1,...r}，f≠g且D'= Df U Dg,对任意i∈{1,...,n}，k∈{1,...,s}，存在：





其中k'是群集D'的序列号。

**3.2一致度和聚合度**

在传统粗糙集的情况下，一致的决策表和不一致决策表仅仅被给出，定义决策表一致的措施并没有被给出。事实上，在一个集群决策表（U，AU{D}），对于每个ak∈A，Mk的第j列表示域中每个对象的成员分布，相当于聚类模型P中的集群Dj。本文认为，这种分布可以包含条件属性集A和决策表聚类模型P之间协调性的信息，进一步反映了聚类精度可以用作衡量聚类过程的一个标准。在以上分析的基础上，我们提出了一致度定义。

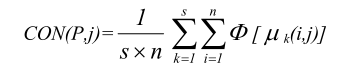
**定义3.2**命名映射Φ：[0,1]→[0,1]作为协调度，若满足下列性质：

1. Φ(x)=1，当且仅当x=0或1，且Φ(1/2)=0时
2. Φ(x)=Φ(1-x)
3. Φ(x)在[0,1/2]上单调减少

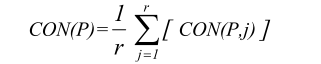
在定义3.2，条件（1）表明，当自变量x是1或0，即对象完全或不完全落下指定的群集，协调度能达到最大值1；相应地，当x为1/2，即成员关系完全不清楚，使得协调度达到最小值0；条件（2），表示该协调度是大约x=0.5对称性；条件（3）指定映射的单调特性。

我们引进Φ(x)=2×|x-0.5|作为协调功能。

**定义3.3**决策表(U,A U{d})，P=U/R{d}={D1 ,...,D r }。我们将P中的Dj的一致度定义为：

(1)

定义决策表的一致度为：

(2)

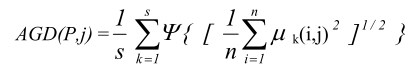
明显，CON(P,j),CON(P)∈[0,1]。

**定理3.3** 在(U,A U{d})，若P=U/R{d}={D1 ,...,D r }={U}，那么CON(P)=1。

证明：它遵循CON(P)形式定义

不考虑P={U}这种情况，CON(P)的值越大，聚类精度就越高。聚类过程应该使CON(P)向扩展的方向前进。但是通过CON(P)引导聚类是不完整的，这是因为一致性不包括P中包含对象信息的集群。本文引出聚合度的概念表明集群含有对象的程度。

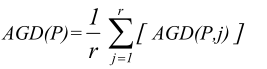
**定义3.4** 决策表(U,A U{d})，n=|U|，A={a1 ,..., as }，P=U/R{d}={D1 ,...,D r }，我们定义P中Dj的聚合度为：

(3)

其中



且决策表的聚合度定义为：

(4)

在定义3.4中，平均平方根用来定义AGD(P,j)，而算术平均值未被采用。这主要是由于避免产生琐碎值（即简单0或1）。该方法借用文献[9]中列出的预测度定义方法。但预测性和聚合度有三点差异：第一是，预测度由概率分布来定义，而聚合度由成员矩阵来定义；第二是，聚合度已归一化处理，从而避免随着集群数量的增加聚合度能迅速达到相当大的数值;第三个是使用Ψ函数，方便匹配一致度。至于Ψ函数，本节将在后面详细分析。

**定理3.4**决策表(U,A U{d})，P=U/R{d}={D1 ,...,D r }，则AGD(P)∈(0,1]。

证明：对任意ak∈A，由于µk(i,j)∈[0,1],则 AGD(P)≤1。因为如果Mk的任意列元素都为零，这表明U中任何对象不属于此列对应于P中的集群下，这是不可能的，所以我们可以得到AGD(P)>0。综上所述，AGD(P)∈(0,1]成立。

**定理3.5**决策表(U,A U{d})，P=U/R{d}={D1 ,...,D r }，且D'=Df U Dh，任意f，h∈{1,...r}，f≠h。若P'=(P-{ Df ,Dh })U{D'}，那么AGD(P')>AGD(P)。

证明：它遵循定义3.4和合并定理。

依据定义3.4和定理3.5，AGD(P)表示P中集群对象的含有度。在凝聚层次聚类算法中，从初始含有n个集群的P，即每个对象是一个集群，到最后只包含一个集群的P，即所有对象组成一个集群，形成了动态聚类图。在动态聚类过程中，聚合度单调递增，但一致度不一定。实际上，在实际应用中一致度在聚合度增加时会有一定程度的下降。聚合度单调增加的性质具有重要的应用价值。我们可以指定阈值λ∈（0,1）的聚合度作为算法的终止条件。一个有效的算法应该综合考虑CON(P)和AGD(P)两个指标，从而使CON(P)在聚类过程中增加，或是在AGD(P)增加时轻微的减少，为此介绍了聚类水平计算公式。

**定义3.5**决策表(U,A U{d})，P=U/R{d}={D1 ,...,D r }。我们将P中的Dj的聚类水平定义为：

(5)

决策表的聚类水平定义为：

(6)

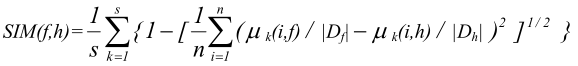
聚类水平可全面反映一致度和聚合度的两个指标。这是很容易证明LEV(P)是关于AGD(P)和CON(P)的单调递增函数。显然LEV(P) ∈ [0,1]，当且仅当CON(P)=0时，LEV(P)=0；当且仅当CON(P)=AGD(P)=1，LEV(P)=1。从定义3.5中可知AGD(P) 和CON(P)在LEV(P)中是两个对称参数。当AGD(P) 和CON(P)的值很接近时，它们对LEV(P)会产生相似的影响；当一个参数的值比其他参数要小得多，参数小的值将对LEV(P)有更大的影响。

Ψ函数引入AGD(P)的定义（定义3.4）。这是因为，在凝聚算法的初始阶段，AGD(P)的值较小时（或当数据集非常大显得非常小），这样引入函数Ψ的主要目的在于在算法初始阶段提高AGD(P)的值以便与CON(P)匹配。而不引入函数Ψ算法的初始阶，AGD(P)在聚类水平的值太小使得CON(P)发挥作用，这无疑是影响聚类算法的精度的。

**3.3基于欧氏距离的分类相似性度量**

在本节中，提出了一种新的分类相似性度量。

**定义3.6** 在决策表(U, A U{d}),A={a1 ,..., as }， P=U/R{d}={D1 ,...,D r }，n=|U|，∀ f, h∈{1,...,r}，P中两个集群Df和Dh之间的相似度可定义如下：

 (7)

定义3.6引用欧氏距离来描述集群之间的相似性，这样的距离越大，相似的值越小。不仅可以利用欧氏距离定义的相似性测量两个集群间同属性的书目和差异属性数目上的不同，而且关键在于表达相似属性和不同属性的程度，其本质是对分类属性进行数值处理。在公式中，µk(i,f)/|Df|和µk(i,h)/|Dh|分别表示集群Df和集群Dh的中心，即用成员平均值表示聚类中心这样一种方式，借用了著名的k-均值算法[1]在数值聚类中的表达方法。若Df和Dh包含一个对象，那么|Df|=| Dh |=1。因此，定义3.6也适应这种情况，所以，定义3.6表示集群与集群之间、对象与集群以及对象与对象的相似性。

**3.4算法**

本节介绍了基于粗糙集层次凝聚聚类算法（RACHA）。该算法首先令P=U/R{d}={{x1},...,{xn}}，然后根据聚类水平和相似性度量进行合并操作。该操作可以一直进行直到达到用户指定的集群数m或者聚合度阈值λ并输出聚类结果，即聚类模型P。算法也可以一直进行直到所有的对象合并为一个簇。在这种方式中，该算法输出动态聚类图。

**算法RACHA**

输入：数据表（U，A），集群数m（或聚合度阈值λ）

输出：聚类模型P

步骤1：令P(0)= {D1 ,...,Dr }={{x1},...,{xn}}, n=|U|, r=n。

步骤2：根据定义3.1推导成员矩阵Mk，k∈{1,...,|A|}。

步骤3：重复以下步骤直至r=m（或者LEV(P)≥λ）

步骤3.1 ：运用公式（5）找到LEV值最小的集群Dmin

步骤3.2：运用公式（7）比较Dmin和P中其它集群的相似度，将与Dmin相似度最高的集合定义为Dsim

步骤3.3：通过根据合并定理合并Dmin和Dsim来更新聚类模型P和成员矩阵Mk，k∈{1,...,|A|}，然后r=r-1

该算法的复杂度分析如下。在步骤2中，为了计算Mk，先要计算U/R{ak}，且Mk中的r×|U|元素需要被控制，其中r是P中|U|值最大集群的数目。文献[14]中的算法1可以用来计算等价类，其时间复杂度为O(|A||U|)log(|U|)，则步骤2的时间复杂度为O(|A||U|log|U|)+O(r|A||U|)。在步骤3中，迭代的最大数目是| U |-1，并且每次迭代需要在步骤3.1中计算LEV r次和在步骤3.2中计算SIMr-1次，它们都可以在O(r|A||U|)内计算，而且步骤3.3的更新操作也可在O(r|A||U|)）完成，这样，步骤3的时间复杂度为O(r|A||U|2)。所以算法的时间复杂度可以估算为O(r|A||U|3)。我们可以很容易地分析得到空间复杂度为O(r|A||U|2)。值得一提的是，上述分析的是最不利的情况，并视Mk为稠密矩阵，但在实际应用中Mk是高度稀疏的矩阵。使用一种适当的稀疏矩阵操作策略，算法的时间复杂度和空间复杂度将显著降低。

1. **实验结果**

**4.1人工数据集**

在这一部分中，我们用一个简单的例子来展示RAHCA的执行。在聚类决策表1中，域为U={x1,...x5}，初始分类条件属性集A={a1, a2 ,a3}，d是引入决策属性域Vd={1,...,5}，而且它是在它的初始状态，如表1：

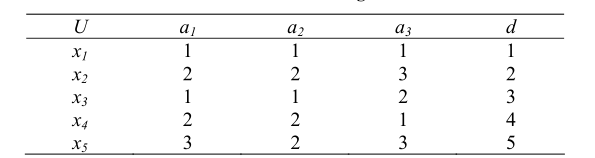


表1 聚类决策表

在第一次迭代，Dj 的LEV值, j∈{1,...,5}分别为0.523，0.515，0.591，0.515和0.581。LEV[P(0),2]=0.52是最小值，让Dmin=D2，D2和其它对象之间的相似值分别为0.57，0.53，0.85和0.82，很明显最相近的簇是D4，SIM(2,4)=0.85。通过合并D2和D4，可以得到 P(1)={x1,{x2 ,x4},x3,x5}。在相同的计算条件，第二迭代合并D1和D3，可以得到P(2)={{x1 ,x3},{x2 ,x4},x5}。在最后一次迭代，P(4)={{x1,x2,x3,x4 ,x5}}，因此，算法结束。分析结果表明：P(2)和P(3)是合理的结果。

动态聚类图如图1所示，而表明CON(P), AGD(P)和LEV(P)在每一次迭代过程中数值变化的曲线如图2所示。

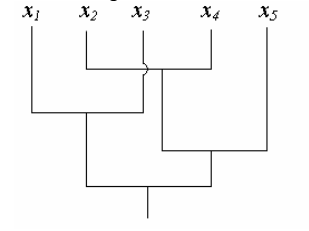


图1 动态聚类图

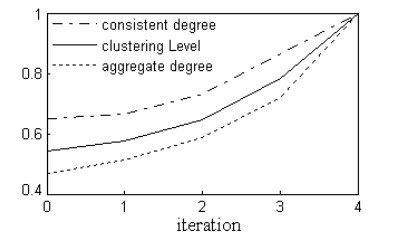


图2 各次迭代中的一致度、聚集度和聚集度值

* 1. **UCI数据集**

我们实验了从UCI机器学习库中获得的3个数据集。气球数据集包含20个实例，每个实例有4个分类属性。它被分为两大类。大豆数据集中含有47个大豆病害的实例，每个实例都有35个分类属性。该数据集根据大豆疾病类型将其分为4类。投票数据集从1984美国国会投票记录数据库提取。它包含435个实例，分别代表267名民主党人和168名共和党人的投票记录。每个实例有26个布尔属性。在日期设置中有一些缺失的属性值，它们被视为一个特殊的常量值。

我们分别采用了在文献[2,9]中提出的算法和本文提出的RAHCA算法（算法A，算法B和算法C为他们的缩写形式）。实验结果如表2所示。在实验中的集群标准的数字是指定的。一个算法的精度是在第i个集群及其相应的实现类的实例的数据集的数量发生的实例总数的比。A算法的结果是迭代50轮的平均准确值。实验结果表明，算法C，即RACHCA，对以上数据集的精度最高。

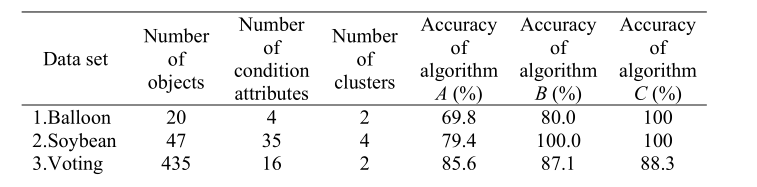


表2 实验结果

1. **结论**

本文将粗糙集应用在数据挖掘的聚类分析中，并介绍了决策属性来配制集群决策表，由此来确定成员矩阵。本文提出了一致度和聚集度指标对应的决策表聚类模型P。这两者分别表示了聚类过程的两个方面。一致度表示关于条件属性集和决策表聚类模型P的集群的等价类之间的协调度，而聚合度表示簇本身时的对象的含有程度。为了综合考虑这两个因素对聚类过程的影响，设计了一个聚类层次的计算公式，其中的一致度和聚合度都处于对称位置，从而使聚类过程能够同时兼顾两个指标。当它们的值是接近的，它们可以影响聚类的水平，当其中一个是较小的，这措施将在聚类水平中发挥重要的作用。正是因为这个原因，它可以成为影响聚类方向的一个主要因素。在实际应用中，初始聚类阶段的聚合度很小，因此聚合度是一个重要的影响算法的因素，在这个时候，算法的聚合度应该提高。随着聚类的不断进行，聚合度逐渐增大，并与一致度一起指导聚类。聚类的后期，一致的程度成为一个主要的影响因素，因为它变得越来越小，因为它在这个时候，算法应该首先保证一致度。

此外，本文提出了基于欧氏距离的分类数据的相似性度量的定义，以便更好地解决非数值数据性质分类数据的困难测量的问题。这项措施适用于集群与集群和集群与对象以及对象与对象之间的比较。此外，在这一方法中，对各参数的操作方便，并使这些参数更加协调、合理地发挥其作用，使其一致度、聚集程度和相似性度量标准化。

基于上述分析结果，本文设计了基于凝聚层次聚类算法的粗糙集（RAHCA）适用于分类数据。该算法需要用户提供的集群数或聚合度阈值。没有这些参数指定，该算法输出的动态聚类图，需要进一步分析，以获得聚类结果。未来的研究工作将包括进一步提高算法效率以及研究关于混合数字和分类数据的聚类算法。

**参考文献**

[1] Mac Queen, J, "Some methods for classification and analysis of multivariate observations", Proceedings of the 5th Berkeley Symposium on Mathematics, Statistics and Probability, 1967, pp. 281-297

[2] Huang, Z., "Extensions to the K-means Algorithm for Clustering Large Data Sets with Categorical Values", Data Mining and Knowledge Discovery Ⅱ. 1998, pp. 283-304

[3] Huang, Z., Ng, M.K., "A Fuzzy K-Modes Algorithm for Clustering Categorical Data". IEEE Trans. Fuzzy Systems, vol.7, 1999, pp.446-452

[4] Bradley, P., Fayyad, U., "Refining initial points for k-means clustering", Proceedings of the 15th International Conference on Machine Learning. Morgan Kaufmann, Los Altos, CA.1998

[5] Sun.Y., Zhu, Q.M., Chen, Zh. X., "An iterative initial-points refinement algorithm for categorical data clustering", Pattern Recognition Letters, vol. 23, 2002, pp. 875-884

[6] Kim, D.W., Lee, K. H., Lee. D., "Fuzzy clustering of categorical data using fuzzy centroids". Pattern Recognition Letters, vol. 25, 2004, pp. 1263-1271

[7] Michalski R., Stepp, R., "Automated Construction of Classifications: Conceptual Clustering Versus Numerical Taxonomy", IEEE Trans. Pattern Anal. Machine Intell., vol. 5, 1983, pp.369-410

[8] Fisher, D., "Knowledge Acquisition via Incremental Conceptual Clustering". Machine Learning, vol. vol. 2, 1987, pp. 139-172

[9] Luis Talavera and Javier Béjar, "Generality-Based Conceptual Clustering with Probabilistic Concepts", IEEE Trans. Pattern Anal. Machine Intell., vol. 23, 2001, pp. 196-203

[10] Guha, S., Rastogi, R., Shim, K., Rock: "A Robust Clustering Algorithm for Categorical Attributes", Proc. Int. Conf. Data Engineering (ICDE'99), Sydney, Australia, 1999, pp. 512-521

[11] Pawlak, Z., "Rough sets", International Journal of Computer and Information Science, vol.11, 1982, pp. 341-356

[12] Pawlak, Z,: "Rough Sets: Theoretical aspects of reasoning about data", Kluwer Academic Publishers, London , 1991

[13] Zhang, W.X., Mi, J. Sh., Wu, W. Zh., "Approaches to Knowledge Reductions in Inconsistent Systems", International Journal of Intelligent System, vol. 18, 2003, pp. 989-1000

[14] Liu Sh. H. et al: "Research on efficient algorithms for rough set methods", Chinese Journal of Computers, Vol. 26, 2003.5, pp.524-529(in Chinese)