吉林大学

本科生毕业论文

中文题目 基于数据时空属性的聚类研究

英文题目 Clustering Research Based on Temporal and Spatial Attributes of Data

学生姓名 李禹恒 班级 一班 学号 19080107

学　　院 电子科学与工程学院

专　　业 电子信息科学与技术

指导教师 付琨\黄宇 职称 研究员\助理研究员

目 录

摘 要

聚类分析是数据挖掘中的一个重要研究领域，作为一种数据划分或分组处理的重要手段和方法，并被广泛应用于天文学、地理学、气象学、地图学及市场分析等众多领域中。

本文的主要工作是对含有时间、空间属性的消息对象进行聚类研究。通过对传统的聚类算法在对时间、空间数据聚类效果的分析讨论，根据其存在的问题进行改进，实验证明改进后的算法在准确率、稳定性以及时间复杂度上都有所提升，同时解决了算法不能自发选择聚类数这一问题，实现了聚类的无监督化。

本文的主要贡献如下：

1) 在地理信息系统表达消息对象的过程中引入聚类分析思想并提出基于多属性数据聚类的层次化思想，解决了含有时空属性的数据的组织问题；
2) 提出基于归一化代价函数的聚类数选择方法、基于深度函数和阈值法的 MinMax 原则以及基于顺序查询的层次化聚类算法以改进传统的聚类算法；
3) 根据本文对于各种算法的研究，自主设计了一款聚类分析软件，用以对各种聚类算法进行聚类实验并对含有时空属性的样本进行层次化联合聚类演示，为在地理信息处理系统中引入聚类分析策略提供了产品原型。

关键词：聚类，时空属性，GIS，聚类数选择

ABSTRACT

Cluster analysis is a major field in data mining, which has been widely applied in such fields including: astronomy, geography, meteorology, cartography and market analysis, as an important method of data partition and data classification.

The primary purpose of this paper is to cluster message subjects with temporal and spatial attributes. After a discussion upon the effectiveness using different traditional clustering methods in clustering data with temporal and spatial attributes, and an analysis upon their existing flaws, a serial of improvements have been proposed. Results of MATLAB experiments shows that improved arithmetics have an better performance on accuracy, stability and time complexity, meanwhile, they've provided some ways leading to select the optimum clustering numbers unsupervisedly.

The main contributions of this paper are enumerated as following:

1) Introducing the clustering methods in the process of expressing message subjects in GIS;
2) Introducing the idea of Hierarchical Theory based on the multidimensional data clustering.
3) Introducing some improved methods such as arithmetic in selecting the optimum clustering numbers based on normalized cost function, the MinMax theory based on Dept-function and threshold and an improved hierarchical clustering method based on Ranking-inquiry.

According to the study upon various of arithmetics, an self-developed cluster analysis application has been designed, which can be used in demonstrating the comparison between different arithmectics and the hierarchical co-clustering towards samples with temporal and spatial attributes. Finally, this application provides a prototype in introducing cluster analysis to GIS.

Keywords: clustering, temporal/spatial attribute, GIS, selection of clustering number

第1章 绪论

第1节 问题的提出

在社会飞速发展的今天，我们身边每时每刻都在产生着数以亿计的消息，这些消息的内容可能千差万别，关乎社会的方方面面。面对这些大量的、不相关的消息，人们无法对其一一进行人工处理，因此需要一种有效的信息处理机制，能够自发的提取海量消息中的各种属性特征，并对其进行聚合分类，从而为对这些信息的识别、检索以及管理提供便利。

根据 L. Shuman 在其书《Practical Journalism》中所述，新闻具有五要素（5W）即何时（when）、何地（where）、何因（why）、何事（what）、何人（who），这是新闻不可缺少的五个方面，是对新闻报道的基本要求[1]。对于消息的组织，我们也可以参考五要素其中的三个属性，即按照消息发生的地域、产生时间或者关键词进行分类。进一步的，这种分类还可以是有序的、层次化的，换而言之，针对消息不同属性的分类，既可以是并列进行的，也可以是顺承进行的，举个例子，全球各地的影院每天都会上映新的电影，一名电影观众如果想要在这些影讯信息中检索到自己关注的信息，往往按照如下顺序进行：首先选择自己所在城市的影院信息，其次要选择当月或未来几个月内的影讯，最后再从不同的影片中进行选择。这样的行为看似随意，实际符合了人类对信息认知的自然规律规律。本文确定了主要研究对象是数据的时间、空间属性，对二者进行有效地信息组织既可以为信息处理的下游提供便利。

本文研究的项目背景依托于一个地理空间信息处理系统。所谓地理空间信息处理系统，是指以特定的地球投影数据模型进行空间定位，对地理空间实体的空间特征信息和属性特征信息进行组织管理、存储查询、空间计算分析、可视化表达输出、专业模型处理和应用的信息系统。这种高度可视化的信息系统被广泛地应用于资源调查、地理教学、城市规划、农林牧业、国土管理、商业金融等几乎所有领域之中[2]。图 1- 1 展示了此类系统的代表 Google Earth。

通过地理空间信息处理系统这个平台，我们可以对前文所述的海量消息按照其发生的地点在平台上进行标记，从而在空间维度上对消息进行组织。

然而对于大量的原始消息数据，简单的将它们按照事件发生的地点在地图上

进行标记并不能取得较好的效果，过于庞杂的信息往往会干扰人们从中提取有用信息。

纵观数据库系统的发展历史，在 20 世纪 80 年代末期，人们提出数据挖掘的概念[3]，即从庞大的数据库中挖掘出有效的信息，从而摆脱了“数据丰富，信息贫瘠”这样的困境。

在数据挖掘的诸多方法中，聚类分析占据了举足轻重的地位，因其在机器学习中能够自发的找到数据的潜在规律而被广泛地应用于信息组织的各种实践中。

古语有云“物以类聚，人以群分”，聚类的过程实际上就是按照事物的某些属性，把事物聚集成类，使同一个类中的对象之间具有较高的相似度，而不同类中的对象差别较大。聚类是人类一项最基本的认知活动，通过适当聚类，事物才便于研究，事物的内部规律才可以为人类所掌握。

至此，本文将尝试利用聚类分析的思想，对地图上以及时间轴上的消息点的时空属性进行聚类研究，找到其时空上的关联，将其聚合成为较少的有代表性的聚点，为数据挖掘下游的工作服务。同时在针对算法的研究中，通过对现有聚类算法的合理改进，以及对多维属性聚类样本的信息组织的深入讨论，使传统聚类算法更贴合于本研究的应用要求，更加准确高效地实现了对海量信息聚类问题。

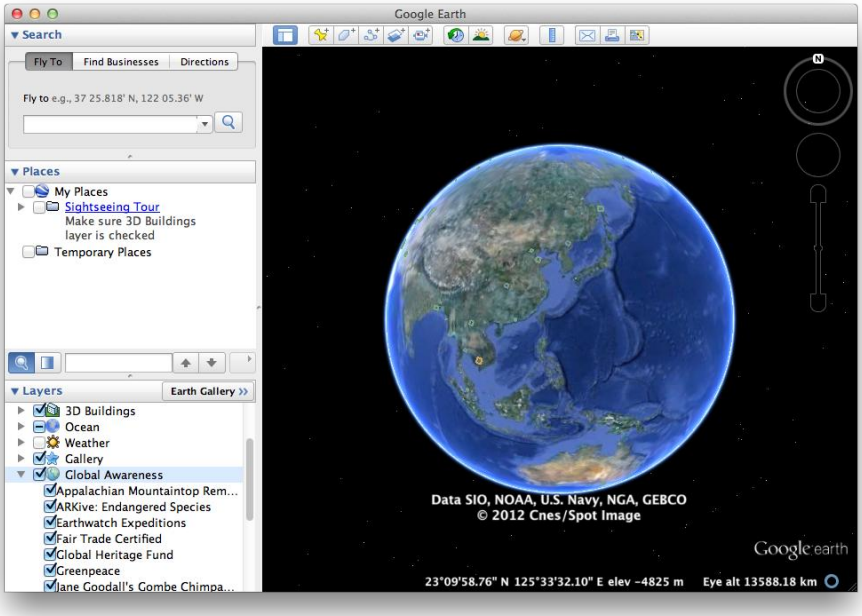


图 1- 1 Google Earth 软件

第2节　研究思路

根据第 1 节的讨论，确定本文的研究思路如下（参考图 1- 2）：

1) 首先一条完整的消息中应该包含有消息产生的时间，发生的地点以及消息的主要内容。因此首先要对消息中的时间空间属性进行提取（该问题不在本文研究范围之内）；
2) 经过时空信息提取之后的消息成为一个含有时间、空间两个属性的样本，本文将基于这些样本进行聚类研究，故可称其为聚类样本；
3) 对于样本的空间属性，采用基于划分的 K-means 算法以及基于模型的 GMM 算法进行聚类分析研究，总结其优势与不足，并针对传统算法需要提供聚类数以及随机选择初始聚类中心等问题，分别利用距离代价函数和 MinMax 原则对其进行改进；
4) 对于样本的时间属性，采用凝聚层次聚类方法对其进行聚类分析研究，总结其优势与不足，并针对凝聚层次聚类方法时间复杂度高的问题，采用顺序查询原则对其进行改进；
5) 针对含有多个不同维度属性的聚类对象的聚类方法以及结果的信息组织等问题提出层次化聚类的策略，用以监督对不同属性的聚类过程及结果表达；
6) 得到的聚类结果可供数据挖掘下游的环节提供参考；

在对各种算法进行分析和改进的过程中，通过大量的实验来发现问题，对比结果，总结算法有效性和准确性，同时为了对本文讨论过的各种聚类算法进行聚类实验并对含有时空属性的样本进行层次化联合聚类演示，本文设计了一个聚类算法测试软件，实现了包括数据导入，算法选择，结果表示等一系列的功能。

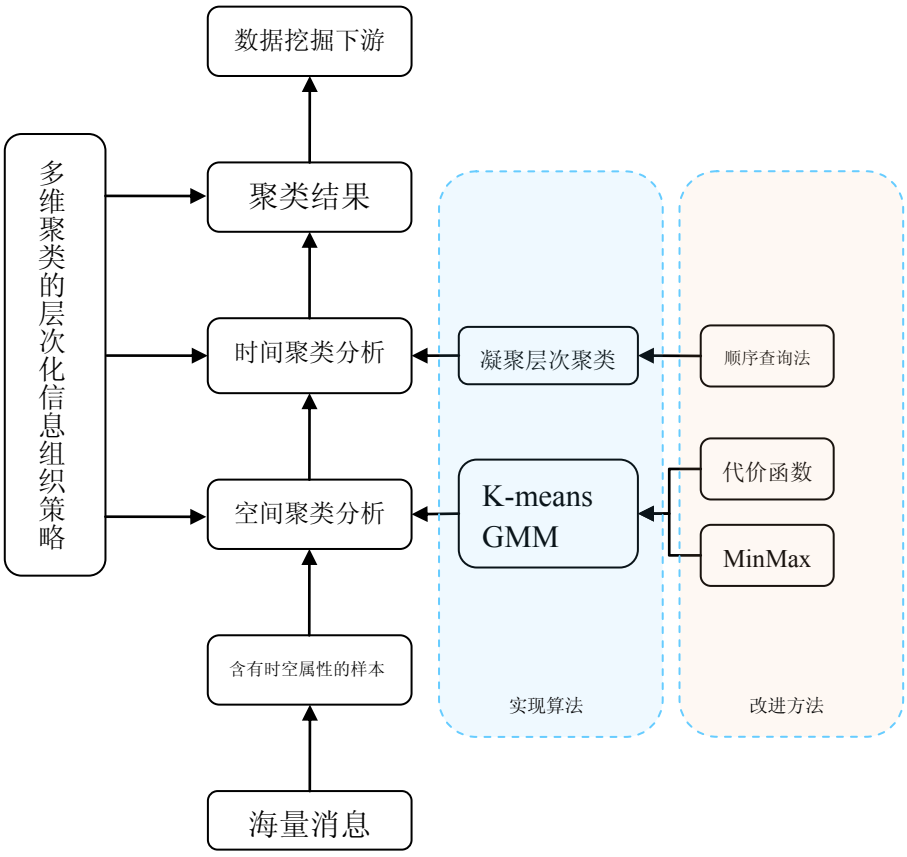


图 1- 2 研究思路框图

第3节 主要贡献与创新点

在前人基于传统聚类算法研究的基础上，本文主要在以下方面进行了深入讨论和创新性研究：

提出了基于归一化代价函数和 MinMax 原则的最优聚类数选择方法，解决了传统 K-means 算法及 GMM 算法不能自动计算聚类数最优解的问题。

提出基于顺序查询的层次化聚类算法，解决了层次聚类时间复杂度较高的问题。

提出基于多属性数据聚类的层次化思想，解决了含有时空属性的聚类对象的信息组织问题。

本文还针对改进的算法自主设计了一款聚类算法测试软件，实现了基于不同算法的聚类演示及对含有时空属性的样本进行层次化联合聚类分析。

第4节　本文的组织结构

本文共分五章，各章内容如下：

第一章：论文问题的提出，研究思路，主要贡献及创新点；

第二章：对支撑本文研究工作的理论进行概述；

第三章：基于数据时空属性，广泛地讨论了基于划分、基于模型以及基于层次的多种不同的聚类算法，详细阐述算法的主要思想，对其进行分析，并针对其存在的不足进行改进。

第四章：本章主要探讨针对数据时空属性的信息组织及表达策略，以及聚类算法测试软件的设计。

第五章：对本课题的研究工作进行总结，并提出有待进一步研究的问题。

第2章 聚类分析综述

本章首先介绍了数据挖掘的概念和步骤，从中说明聚类分析作为数据挖掘的重要环节，进而介绍了聚类分析的内容、分类，根据本文研究的对象介绍了聚类分析中的相似性度量准则，最后给出聚类算法的有效性评价方法。

第1节 数据挖掘概述

数据挖掘（Data Ming）被认为是一种将数据转化为有用信息和知识的重要途径。简单来讲，数据挖掘就是从大量的、不完全的、有噪声的、模糊的、随机的实际应用数据中提取隐含在其中的、人们事先不知道的、但又是潜在有用的信息和知识的过程。它既是信息进化的产物，又是知识发现（图 2- 1）的重要环节[4]。

数据挖掘通常因循如下步骤：

1. 问题的定义：数据挖掘是为了从大量数据中发现有用的令人感兴趣的信息，因此发现何种知识就成为整个过程中的第一个，也是最重要的一个阶段。在整个过程中，必须确定数据挖掘所需要采用的具体方法；
2. 数据收集与预处理：主要包括数据选择、数据与处理以及数据转换，其目的在于消除噪声、遗漏数据处理、消除重复数据数据类型转换从而从源数据中抽取与挖掘任务相关的数据集，从而提高数据挖掘的效率；
3. 数据挖掘的实施：通过分类、聚类、关联的过程对上一步得到的有用数据集进行挖掘；
4. 结果解释与评估：实施数据挖掘所获得的结果，需要进行评估分析，以便有效发现有意义的知识模式。

数据挖掘中常使用如下分析方法：

1. 分类：首先从数据中选出已经分好类的训练集，在该训练集上运用数据挖掘分类的技术，建立分类模型，对于没有分类的数据进行分类。
2. 估计：与分类类似，不同之处在于分类描述的是离散型变量的输出，估值处理连续值的输出，且估值量不确定。
3. 预测：根据分类或估值得出模型，利用该模型对未知变量进行预测。
4. 聚类：对记录分组，把相似的记录在一个聚集里。聚类和分类的区别是

聚集不依赖于预先定义好的类，不需要训练集。

图 2- 1 数据挖掘是知识发现过程中的一步

第2节　聚类分析概述

在数据挖掘的诸多方法中，聚类分析占据了举足轻重的地位，因其在机器学习中能够自发的找到数据的潜在规律而被广泛地应用于信息组织的各种实践中。本节将系统地归纳聚类分析的基本概念及聚类方法，讨论聚类分析对象的数据类型和数据结构，确定聚类的相似度度量准则以及有效性评价方法。

2.1 聚类分析概述

聚类是人类一项最基本的认知活动，通过适当聚类，事物才便于研究，事物的内部规律才可能为人类所掌握。古语有云“物以类聚，人以群分”，聚类的过程实际上就是按照事物的某些属性，把事物聚集成类，使同一个类中的对象之间具有较高的相似度，而不同类中的对象差别较大。具体来讲，聚类分析的输入可以用一组有序对(X，s)或(X，d)表示，这里 X 表示一组样本，s 和 d 分别是度量样本间相似度或相异度。聚类系统的输出是一个分区 $C=\{C_1，C_2，\ldots，C_k\}$，其中 C_i 是 X 的子集，称为类或簇[5]。

聚类是一个无监督的学习过程，它同分类的根本区别在于：分类是需要事先知道所依据的数据特征，而聚类是要找到这个数据特征，因此，在很多应用中，聚类分析作为一种数据预处理过程，是进一步分析和处理数据的基础[6]。

作为数据挖掘领域中一个非常活跃的研究领域，聚类分析被广泛应用于统计学、机器学习、空间数据库、生物学以及市场营销等领域[7]。

2.2 聚类算法分类

目前传统的聚类算法可以被分为五类：划分方法、层次方法、基于模型方法、基于密度方法和基于网格方法。[8][9]

1) 层次的方法（Hierarchical Method）

基于层次的聚类方法的本质是对给定的数据对象集合进行层次分解。按照层次的形成方式，层次方法可以分为凝聚的方法和分裂两种方法[10]。凝聚的方法，也称为自底向上的方法，一开始将每个对象都作为单独的一个类，然后按照某种相似性评价准则，将相似度最高的若干对对象在新的一层中合并成同一类，通过反复迭代，直到所有的类合并成一个(层次的最上层)，或者达到一个终止条件，迭代终止。分裂的方法，也称为自顶向下的方法，一开始将所有的对象都置于同一个类中，然后通过不断的迭代，在迭代的每一步中，一个类按照某种相似性评价标准，被分裂为更小的类，最终直到每个对象被归入某个单独的类中，或者达到某个终止条件为止。

2) 划分的方法（Partitioning Method）

基于划分的聚类方法的本质是将给定的数据对象集合划分到指定个数个划分区域中，每个区域作为一个类，算法会按照一个客观的分类准则反复验证，对给定数据对象进行重新分配，直到条件被满足为止。

典型的算法有 k-means 和 k-medoids[11]。其中 k-means 方法采用采用类内对象的均值作为聚类中心，通过计算每一个对象与划分出的各类中心的相似性来决定该对象属于哪一类，通过计算类内对象均值与聚类中心的偏差来决定算法停止。K-medoids 算法则选取类中最接近中心的实际的对象来代表该类，与 k-means 相比，k-medoids 对噪声不敏感，但复杂度较高，两种算法都需要指定聚类数 k。

3) 基于密度的方法（Density-based Method）

基于密度的聚类方法的主要思想是将临近区域密度（对象或数据点的数目）超过某个阈值的对象聚为一类，换而言之，对给定类中的每一个数据点，在一个给定范围的区域中必须至少包含一定数目个同类数据点。基于密度的方法为算法对噪声的敏感提供了解决办法，同时不像绝大多数聚类方法基于对象之间的距离进行聚类（这样只能发现平面上圆形或空间中球状的类），而是可以发现任意形状的类（只要该形状的密度足够大）。

4) 基于网格的方法（Grid-based Method）

基于网格地方法把对象空间量化成有限数目个单元，形成一个网络结构。所有的聚类都是基于这个网络结构进行的。这种方法的突出优点在于处理速度快，因为它的处理时间与数据对象的数目无关，只与量化空间中每一维的单元数目有关。

5) 基于模型的方法（Model-based Method）

基于模型的方法为每一个类建立一个模型，算法主要是利用数据对给定模型进行最佳拟合。例如本文中用到的高斯混合模型，即通过建立若干个高斯概率密度函数，采用最佳可能性估测法将数据对象指定给不同的高斯概率密度函数，从而达到聚类效果。一个基于模型的算法可能通过构建反应数据点空间分布的密度函数来定位聚类，利用统计数字自动决定聚类数目吗，兼顾噪声数据和孤立点，从而产生健壮的聚类算法。

2.3 聚类分析中的相似性度量

相似性的度量是聚类算法的重要因素，它为聚类算法提供聚类依据。针对不同的数据对象、不同的应用场合，需要提出不同的相似性度量标准。根据本文所要研究的主要对象——消息的空间属性、时间属性以及文本信息，定义了如下聚类统计量，用作聚类分析的度量指标，从而可以定量进行聚类分析。

1) 时间间隔

对于数据对象的时间属性，本文采用一个 12 位整数进行记录，其格式如 YYYYMMDDhhmm，其中 YYYY、MM、DD 分别代表消息产生的年月日，hh、mm 代表具体小时（24 小时制）和分钟。例如 2012 年 5 月 1 日 14 点 07 分，则表示成 201205011407。

针对时间向量$\vec{t}(i)$,定义了时间间隔:

$$dt(i,j) = \left|\vec{t}(i) - \vec{t}(j)\right| \qquad \text{公式（2-1）}$$

2) 空间距离

对于数据对象的空间属性，可以通过计算对象间的距离来衡量其相异程度。最常用的距离度量方法是欧几里得距离：

$$ds(i,j) = \sqrt{|x_i - x_j|^2 + |y_i - y_j|^2} \qquad \text{公式（2-2）}$$

其中 x_i 表示第 i 个对象的横坐标（对应地理经度），y_i 表示第 i 个对象的纵坐标（对应地理纬度）。

3) 余弦相似度

对于数据对象的文本属性，可以用余弦相似度来定义其相似程度，任何两个文档向量 A、B 之间的余弦相似度可以用公式

$$\mathrm{sim}(A,B)=\frac{\sum_{j=1}^{n}\omega_{Aj}\,\omega_{Bj}}{\sqrt{\sum_{j=1}^{n}\omega^{2}{}_{A_j}}\sqrt{\sum_{j=1}^{n}\omega^{2}{}_{B_j}}} \qquad \text{公式（2-3）}$$

表示，其中 n 为特征空间的维数。

对于具有多个属性的聚类对象的聚类研究，了解基于不同度量准则的距离的定义方法十分重要，因为方法的选择直接影响到簇的划分性质，这就要求我们根据具体应用目的来选择合适的度量准则。例如在空间距离的度量准则的选择上，除了上文提到的欧几里得距离，还有诸如曼哈坦距离、明考斯基距离等许多度量准则，而一般来说，欧氏距离能够满足大多数聚类的要求，用此距离度量准则聚类所得到的结果也符合自然解释。

2.4 聚类算法有效性评价

聚类算法的质量通常取决于三个因素，即算法所使用的相似性度量准则、算法实现过程及方法，以及算法能否发现隐藏于聚类对象之中的模式。

在此我们引入 Rand 系数[12]作为聚类算法有效性的衡量准则。该准则规定将聚类结果与真实划分情况比较，对每一个样本对，存在四种可能：

1) 他们应归入一类，结果中确实将其归入一类；

2) 他们应归入一类，结果中将他们分入不同类；

3) 它们不应归入一类，结果中却将它们归入一类；

4) 它们不应归入一类，结果中确实将他们归入不同类。

设满足以上条件的样本对数分别为：a，b，c 和 d，总样本对数为 n。评价标准常采用正确的样本对数与总样本对数之比（Rand 系数）：

$$\mathrm{Rand}=\frac{(a+d)}{n\times(n-1)/2} \qquad \text{公式（2-4）}$$

然而，由于聚类是一种预先不知道据类对象划分情况的过程，因此 Rand 系数评价法只能作为聚类有效性的参考，类的大小没有准确的规定，可以根据据类对象的分布情况而自动适配，亦可以根据聚类的目的，人为规定，因此对于聚类

算法有效性的评价，应在客观的 Rand 系数评价法的辅助下，结合主观上的实际应用目的，给予综合考量。

第3章 基于数据时空属性的聚类算法及改进

本章重点讨论不同聚类算法对数据时空属性进行聚类分析的可行性以及聚类效果，并提出改进方法。首先概述基于划分、模型以及层次的聚类方法的基本思想，通过分析这些传统算法的优势和不足，结合课题的研究目的，在第 2 节、第 3 节以及第 4 节分别利用距离代价函数、MinMax 原则以及顺序查询方法对传统算法进行改进，最后通过实验对改进后算法的有效性进行分析。

第1节　传统聚类算法分析

1.1 基于划分的聚类算法

K-means 算法是基于划分的聚类算法中最为经典的一例，本文将根据 K-means 算法的特点，采用欧氏距离作为相异度度量准则，对其对数据空间属性聚类的可行性及聚类效果进行讨论。

1.1.1　K-means 算法基本思想

K-means 算法是 Mac-Queen[13]提出的一种非监督实时聚类算法，是基于划分的聚类算法中最为常用的一种算法，在最小化误差函数的基础上将数据划分为 K 个预定的簇数[14]。

划分方法的基本思想是将含有 n 个样本的数据集，利用一定的算法将其划分为 k 个划分（K≤n），每一个划分作为一个类（或簇），同时满足：

1)　每个簇至少包含一个样本；

2)　每个样本必须属于且仅属于一个簇。

K-means 算法的基本思想是随机选取 k 个样本作为初始聚类中心，通过迭代将数据对象划分到不同的簇之中。根据聚类性能指标最小化原则，K-means 算法通常使用每类数据样本的均值与该类中心的误差平方和作为聚类准则函数：

$$E = \sum_{l=1}^{k} |p_l - m_l|^2 \qquad \text{公式（3-1）}$$

其中 p_l 表示第 l 类数据对象的均值，m_l 是第 l 类的聚类中心（p 和 m 都是多维的）。

当聚类准则函数足够小（满足一定阈值）时，迭代结束。算法的具体步骤如

下：

1) 算法接受参数 K 作为聚类个数，并随机指定 K 个点作为聚类中心；
2) 通过迭代法计算每一个点到每一个中心的距离，将该点分配给距离最近的那个簇；
3) 计算每一个类中横纵坐标的均值，与该类中心计算误差平方和（公式 2-1），如果误差大于已设定的阈值，则选取该均值作为新的聚类中心，重复 2)；
4) 满足条件，即迭代收敛于某一组稳定的聚类中心，则停止迭代；
5) 输出聚类中心。

算法的基本流程如图 3- 1 所示[15]。可以看出，K-means 算法的输入是聚类个数 k，输出为已划分的各个类（包括聚类中心以及每个据类对象所属类别）。

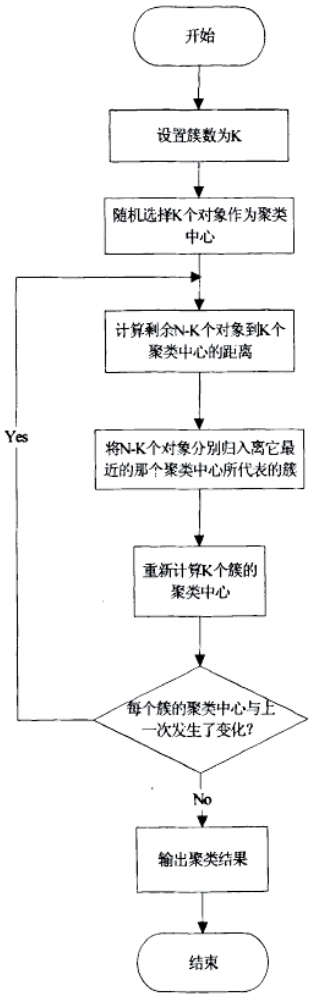


图 3- 1 K-means 算法流程图

1.1.2 K-means 算法分析

K-means 算法的时间复杂度是 O（mnkt）,其中 m 是样本的属性个数，n 是样本个数，k 是聚类个数，t 是迭代次数，通常情况下 k<<n 且 t<<n。比较其他聚类方法，基于划分的 K-means 算法计算速度快，适合处理大规模的数据，且聚类结果紧凑，簇与簇之间的距离明显。经过分析，传统的 K-means 算法具有以下不足[1][16][17]：

1) 需要提前确定聚类个数 K，即不能动态地增加或合并新类。这一问题与本文的研究目的相冲突，为了要充分利用聚类算法的非监督特性，进行聚类的前提是不清楚聚类对象的具体分布情况，人为设置聚类个数不但无法实现数据自发组织这一目的，而且干扰真实结果；
2) 初始聚类中心的选择对结果有严重影响。由于上一小节中描述算法步骤中的第 2 步（见第 13 页）仅仅围绕聚类中心进行迭代，即当第 4 步收敛时，得到的仅仅是一个局部最优解，无法满足全局最优（见图 3- 2）；
3) 算法对噪声敏感。由于聚类中心的选取依赖于均值，故脏数据的干扰会对聚类中心的选取有较大干扰。

根据以上分析，算法急需解决聚类数自动适配以及提供合适的初始聚类中心这两个问题。

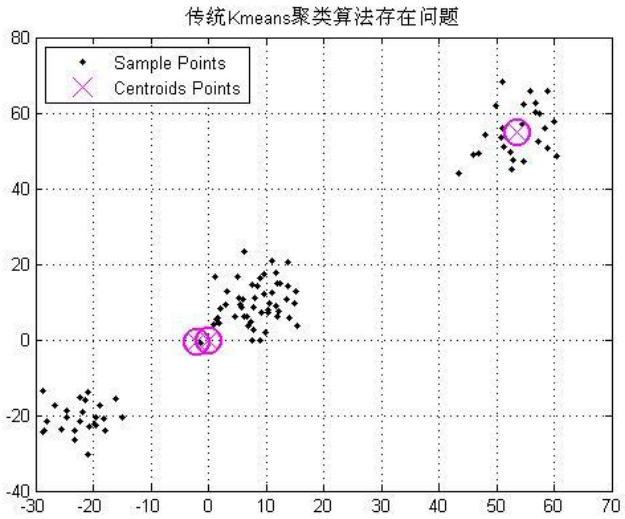


图 3- 2 传统 K-means 算法存在问题

1.2 GMM 聚类算法

GMM 聚类算法是一种基于模型的聚类算法，根据本文的研究目的，在对空间坐标进行聚类的实践中，可以尝试将满足同一高斯分布的若干相近点划分为一类，利用高斯混合模型可以计算出样本中各点分别属于哪几个高斯分布，并能够得到其数学期望，作为样本的聚类中心。

1.2.1 高斯混合模型的基本思想

高斯混合模型（Gaussian mixture model,简称 GMM）是单一高斯概率密度函数的延伸，能够平滑地近似任意形状的密度分布凝聚层次聚类算法[18]。

假设我们有一组在高维空间（维度为 d）的点$x_i, i = 1 \dots n$，若这些点的分布近似椭球状，则我们可以用高斯密度函数$g(x_i, \mu, \sum)$来描述产生这些点的概率密度函数：

$$g(x_i, \mu, \sum) = \frac{1}{\sqrt{(2\pi)^d |\sum|}} \exp\left[-\frac{1}{2}(x-\mu)^T \sum^{-1} (x-\mu)\right] \quad \text{公式（3-2）}$$

其中μ代表此密度函数的中心点，$\sum$代表此密度函数的协方差矩阵（Covariance Matrix），这些参数决定了此密度函数的特性，如函数形状的中心点、宽窄及走向等。欲求得最佳的参数来描述所观察到的数据点，可由最佳可能性估测法的概念来求得。

如果聚类对象分布不属于同一个高斯分布，我们需要采用多个高斯函数的加权平均来表示。假设若以三个高斯函数来表示，则可表示为：

$$p(x) = \alpha_1 g(x; \mu_1, \sum_1) + \alpha_2 g(x; \mu_2, \sum_2) + \alpha_3 g(x; \mu_3, \sum_3) \quad \text{公式（3-3）}$$

此概率密度函数的参数为（$\alpha_1, \alpha_2, \alpha_3, \mu_1, \mu_2, \mu_3, \sum_1, \sum_2, \sum_3$），其中：

$$\alpha_1 + \alpha_2 + \alpha_3 = 1$$

此种方式表示的概率密度函数成为“高斯混合密度函数”或“高斯混合模型”，简称 GMM。

为了简化讨论，通常假设各个高斯密度函数的协方差矩阵为：

$$\sum_j = \sigma^2_j I = \sigma^2_j \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}, j = 1,2,3 \quad \text{公式（3-4）}$$

此时单一的高斯密度函数可表示为：

$$g(x_i,\mu,\sigma^2)=\frac{1}{\sqrt{(2\pi)^d\sigma^2}}\exp\left[\frac{-(x-\mu)^T(x-\mu)}{2\sigma^2}\right]$$ 公式（3-5）

当协方差矩阵可以表示成一个常数和一个单位方阵的乘积时，前式 p(x)可简化为：

$$p(x)=\alpha_1 g(x,\mu_1,{\sigma_1}^2)+\alpha_2 g(x,\mu_2,{\sigma_2}^2)+\alpha_3 g(x,\mu_3,{\sigma_3}^2)$$公式（3-6）

此 p(x)的参数为 $\theta=[\alpha_1, \alpha_2, \alpha_3, \mu_1, \mu_2, \mu_3, {\sigma^2}_1, {\sigma^2}_2, {\sigma^2}_3]$，参数个数为 6+3d。

欲求最佳的 θ 值，可以采用最佳可能性估计法（MLE）原则，求出下列的最小值：

$$J(\theta)=\ln[\prod_{i=1}^n p(x_i)]=\sum_{i=1}^n \ln[\alpha_1 g(x,\mu_1,{\sigma_1}^2)+\alpha_2 g(x,\mu_2,{\sigma_2}^2)+\alpha_3 g(x,\mu_3,{\sigma_3}^2)]$$

公式（3-7）

欲求 J(θ)的最小值，可以直接对 μ_j 及 σ_j 微分：

$$\nabla_{\mu_j}=\sum_{i=1}^n \beta_j\left(\frac{x_i-\mu_j}{{\sigma_j}^2}\right)$$ 公式（3-8）

$$\nabla_{\sigma_j}J(\theta)=\sum_{i=1}^n \beta_j(x_i)\left(\frac{(x_i-\mu_j)^T(x_i-\mu_j)}{{\sigma_j}^3}-\frac{d}{\sigma_j}\right)$$ 公式（3-9）

令上两式为零，即可得到：

$$\mu_j=\frac{\sum_{i=1}^n \beta_j(x_i)x_i}{\sum_{i=1}^n \beta_j(x_i)}$$ 公式（3-10）

$${\sigma_j}^2=\frac{1}{d}\frac{\sum_{i=1}^n \beta_j(x_i)(x_i-\mu_j)^T(x_i-\mu_j)}{\sum_{i=1}^n \beta_j(x_i)}$$ 公式（3-11）

其中，$\beta_j(x)=\frac{\alpha_j g(x,\mu_1,\sigma^2)}{\alpha_1 g(x,\mu_1,\sigma^2)+\alpha_2 g(x,\mu_2,\sigma^2)+\alpha_3 g(x,\mu_3,\sigma^2)}$称为事后概率。$\alpha_j$ 的求法比较复杂，需要满足和为 1 的条件，引进 Lagrange Multiplier，此处只给出结果：

$$\alpha_j=\frac{1}{n}\sum_{i=1}^n \beta_j(x_i), j=1,2,3$$ 公式（3-12）

根据公式 3-10,3-11,3-12，利用迭代法求出 θ，流程如下：

1) 初始化参数 $\theta=[\alpha_1, \alpha_2, \alpha_3, \mu_1, \mu_2, \mu_3, {\sigma^2}_1, {\sigma^2}_2, {\sigma^2}_3]$。令 $\alpha_1=\alpha_2=\alpha_3=1/3$，${\sigma^2}_1={\sigma^2}_2={\sigma^2}_3=1$，并使用 K-means 计算 μ_1, μ_2, μ_3。
2) 使用 θ 来计算 $\beta_1(x_i), \beta_2(x_i), \beta_3(x_i)$，i=1,2，…，n
3) 利用公式 3-10,3-11,3-12 计算新的 μ_j，σ_j，α_j 值，记为$\tilde{\mu_j}$，$\tilde{\sigma_j}$，$\tilde{\alpha}_j$
4) 令$\tilde{\theta}=[\tilde{\alpha}_1, \tilde{\alpha}_2, \tilde{\alpha}_3, \tilde{\mu}_1, \tilde{\mu}_2, \tilde{\mu}_3, \tilde{\sigma}_1, \tilde{\sigma}_2, \tilde{\sigma}_3]$，若$\|\theta-\tilde{\theta}\|$小于某个极

小的容忍值，则停止，否则令$\theta = \tilde{\theta}$并回到步骤 2。

输出结果中的 μ_j 即为待求的聚类中心。

1.2.2 GMM 聚类算法分析

根据上节的介绍，高斯混合模型具有如下优点：

1) 对噪声不敏感。由于所求得的聚类中心是某个高斯分布的期望，该中心必定要存在于数据密集的区域，而噪声点相对稀薄，不会对聚类中心产生影响；
2) 能够识别任何椭球形态的聚类。相比于 K-means 算法只能识别近似正圆型的类，由于 GMM 不曾采用基于欧氏距离的相似性评价标准，凸型的椭球型的类都可以被有效的识别。

然而 GMM 算法同样存在诸如以下问题：

1) 算法需要人工设定聚类数。同 K-means 一样，GMM 算法需要预先提供参数以确定模型中单一高斯分布的个数；
2) 算法受到初始 θ 的影响。上述迭代方法虽然会让 J(θ)逐步递增，并收敛至一个局部最大值，但我们无法证明此局部最大值就是全局最大值；

在一次实验中，虽然已经指定了相对正确的聚类数，但聚类结果（如图 3- 3）依旧与实际情况相差甚远，经过对结果的仔细分析，发现实验结果中每一个聚类仍旧符合高斯分布，之所以会出现箭头所示的病态聚类结果，是因为这两个高斯分布的初始 μ 比较靠近，而这个 μ 是通过 K-means 选取的，根据上一节的讨论，K-means 问题又进一步影响了 GMM 算法的准确性。

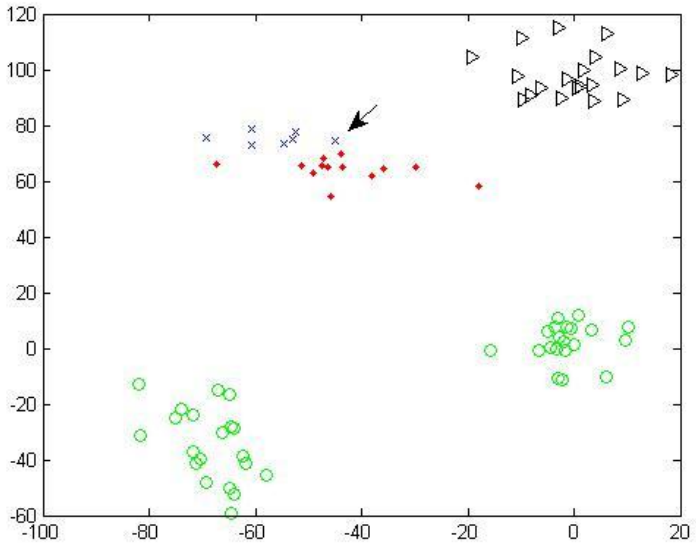


图 3- 3　传统 GMM 算法存在问题

1.3 凝聚层次聚类算法

根据第 2 章 2.3 的讨论，数据的时间属性有较强的层次特征，对于时间属性的聚类，可以尝试利用层次聚类的思想来实现。

1.3.1　凝聚层次聚类的基本思想

层次聚类是一种按照一定方向——或从底部到顶部，或从顶部到底部，遵从一定条件的样本的凝聚或分解。根据本文研究的目的，这里主要讨论的是凝聚层次聚类算法，即根据一定的簇间距离度量准则，将满足一定条件的两个叶节点捏合为一个新的根节点，从而建立起新的一层。算法的实现过程如下：

1) 最底层中 n 个样本各自作为一类；

2) 计算任意两点之间的相异度；

3) 从该层中找出最为相似的两点，由它们形成新的根节点，构成新的一层；

4) 重复步骤 2 直至凝聚成一个根节点，算法结束。

在对数据的时间属性进行聚类的过程中，我们一般选取时间间隔作为相异度的评价标准。

图 3- 4 为一次对于数据时间属性的层次聚类树状图，其数据格式参见第 2 章 2.3 中关于时间间隔的描述，数据样本总数为 120，其数据值所代表的时间集中在 2011 年 2 月、6 月和 9 月以及 2012 年 1 月、3 月、5 月、7 月、8 月以及 10

月，通过树状图可以清晰的看到完整的聚类过程（考虑篇幅，底层聚类过程省略），其中最上层的两个根节点代表着年份，图中最下层表示每个年份下属的月份。

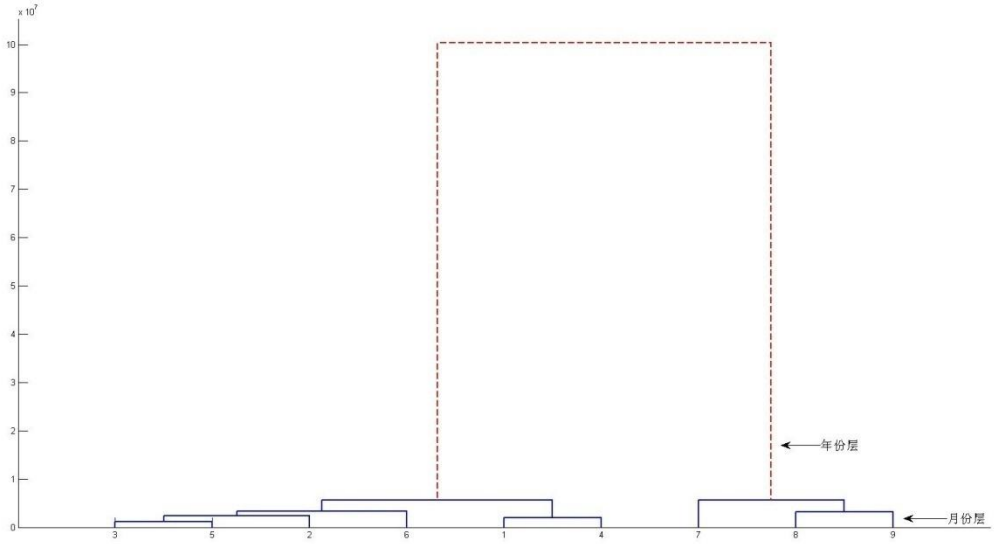


图 3- 4 层次聚类树状图

1.3.2 凝聚层次聚类算法分析

相比于基于划分的方法，层次聚类方法不需要给定其实的聚类中心也可以得到相对准确的聚类结果，不过这种方法仍旧存在如下问题：

1) 算法依旧需要人为给定一个算法结束的标准，那一层的结果最优不在算法的考虑当中，而需要人为设定；
2) 时层次聚类的质量受限于如下特点：一旦一个合并或分裂被执行，就不能修正，这个特点十分容易受到数据的分布的影响，比方说两个不同的类可能由于其边缘距离较近而被视为同一类；
3) 层次聚类算法对于噪声也比较敏感，那些脏数据会使聚类中心向其偏移，或者自成一类；
4) 层次聚类算法在将所有对象合并成一类之前，需要做$\sum_{i=2}^{N} i(i-1)/2$共计（$N^3 - N/6$）次比较大小的操作，而更新距离矩阵时需至少做$\sum_{i=2}^{N}(i-2)$共计（$(N-1)\times(N-2)/2$）次比较大小的操作，因此，层次聚类算法大量的时间都消耗在比较大小的操作上。

因此层次方法仅适合于小型数据集的聚类。

第2节　一种基于归一化代价函数的改进方法

2.1 利用归一化代价函数计算最优聚类数

一个好的聚类的评判标准应该包括（不仅限于）使类与类之间尽可能疏远，而类间各点尽可能的紧凑，据此杨善林[19]等人提出距离代价函数的概念用以记录 K 取不同值时对类际距离与类内距离的影响。

根据杨的研究，它们定义了类际距离、类内距离和距离代价函数（见公式 3-2、3-3、3-4）。

$$L = \sum_{i=1}^{k} |m_i - m| \qquad \text{公式（3-13）}$$

$$D = \sum_{i=1}^{k} \sum_{p \in C_i} |p_i - m_i| \qquad \text{公式（3-14）}$$

$$F(s,k) = L + D \qquad \text{公式（3-15）}$$

其中 K={X,R}为聚类空间，X={x_1,x_2,x_3,⋯,x_n}，聚类数为 k，L 为类际距离，D 为类内距离，F（s，k）为距离代价函数，p 为任意空间对象，m 为全部样本的均值，m_i 为簇 C_i 所含样本的均值。在运用距离代价函数作为聚类有效性检验函数时，根据代价最小准则，当距离代价函数达到最小值时，空间聚类结果为最优，k 的最优选择由下式给出：

$$\min_k\{F(s,k)\}，\ k = 1,2,3 \ldots，\ n \qquad \text{公式（3-16）}$$

令 K={X，R}为聚类空间，其中，X={x_1,x_2,x_3,⋯,x_n}，假设 n 个空间对象被聚为 k 个簇，D 为类内距离，L 为类际距离，当 L=D 时，空间聚类数 k 达到优化，即符合经验规则[1]：$k \leq \sqrt{n}$。

以上定理证明如下：令$\bar{d}$为样本与其聚类中心的平均距离，$\bar{d} = \frac{D}{n}$；$\bar{l}$为聚类中心的平均距离，$\bar{l} = \frac{L}{k}$，当空间聚类具有分形几何特征时，即每个聚类内部的空间结构与整个聚类空间的结构在形态上是相似的时应有：

$$\frac{\bar{l}}{L} = \frac{\bar{d}}{D/k} \qquad \text{公式（3-17）}$$

但是，实际空间聚类不一定具备分形几何特征，考虑问题的一般性，空间聚类应遵循紧致和分离性[20]要求，即一个好的空间聚类应该使各聚类中心的间距

[1]根据经验，n 个聚类对象属于不超过$\sqrt{n}$个类。

尽可能地大，而样本与其中心间距尽可能地小．此时应有：

$$\frac{\bar{d}}{D/k} < \frac{\bar{l}}{L} \qquad \text{公式（3-18）}$$

当 L=D，即$L = k\bar{l} = D = n\bar{d}$时，联立上述（3-6），（3-7）两个方程，容易得到：$k^2 \leqslant n$，即$k \leq \sqrt{n}$。得证。

在对杨的算法进行实验后，发现效果并不理想，主要问题在于：

1) 类际距离受数据的不同空间分布模式影响较大；
2) 当数据较大时，D 和 L 不属于同一个数量级，无法直接比较或者相加；
3) 当数据较小或空间分布较分散时，F 的稳定性很差。

针对第二个问题，可以通过将类际以及类间距离归一化来解决，从而得到归一化的类际距离：

$$\bar{L} = \frac{\sum_{i=1}^{k} |m_i - m|}{\max_i |m_i - m|} \qquad \text{公式（3-19）}$$

以及归一化的类间距离：

$$\bar{D} = \frac{\sum_{i=1}^{k} \sum_{p \in C_i} |p_i - m_i|}{k \times \max_i |p_i - m_i|} \qquad \text{公式（3-20）}$$

继而给出归一化的距离代价函数：

$$\bar{F}(s,k) = \bar{L} + \bar{D} \qquad \text{公式（3-21）}$$

2.2 算法实现

这里给出基于距离代价函数的改进的 K-means 算法的伪代码：

1) 初始化变量，将聚类对象个数赋给 n；
2) 计算全部样本均值 M，计算最大聚类数$k_{max} = \sqrt{n}$；
3) 将 K_{max} 赋给 x 作为聚类数标记，while x>1，do；
4) 进行 k 值为 x 的传统 K-means 算法（参见第 12 页），得到聚类中心 C_x；
5) 利用（2-8），（2-9），（2-10）计算$\bar{F}_x$；
6) x=x-1，重复步骤 4；
7) 循环结束，选取最小的$\bar{F}_x$对应的 x 值作为最优的 k 值，其对应的聚类中心为算法输出结果。

2.3 实验分析与比较

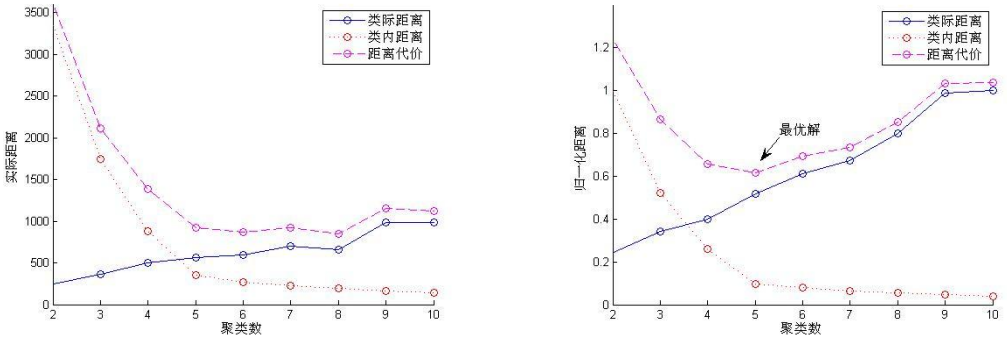
为了给出有参考性的原始测试数据，这里利用 MATLAB 生成了一个含有 5 个满足二维高斯分布的数据集，其均值与方差如表 3- 1 所示。

组别	均值（E_x，E_y）	方差（σ^2_x，σ^2_y）
第一组	（0,0）	（536,536）
第二组	（-110,60）	（180,180）
第三组	（-110，-60）	（133,131）
第四组	（117,59）	（181,181）
第五组	（145，-55）	（181,181）

表 3- 1 测试样本数据

经过 50 次重复实验，用传统的距离代价函数估算 k 值的准确率为 29.41%，改进之后的归一化距离代价函数估算 k 值的准确率达到 64.71%，原算法的准确率和稳定性得到了大幅度的提高。对比图 3- 5 中 A、B 说明改进后类际距离和类间距离的尺度得到统一，代价函数的特征相对明显，代价函数的最低点（即最优解）十分清晰，受数据影响较小。

将类际距离和类间距离归一化，有效地解决了算法对于数据空间分布及数据量敏感的问题，实验表明，归一化之后的距离代价函数可以较稳定的预估出聚类样本的类数，是完成非监督聚类的一个很好的尝试。



A：改进前　　B：改进后

图 3- 5 代价函数改进对比实验

第3节　一种基于 MinMax 原则的改进方法

根据上一章的分析我们已经知道，影响 K-means 算法准确性和稳定性的关键因素在于初始聚类中心的选择，一个基于全局的，相对准确的初始聚类中心为 K-means 算法提供了较好的划分依据，在此基础上，K-means 算法所得到的结果就是全局的，稳定的，而避免陷入局部最优而忽视全局的困境。

基于以上考量，提出了一种基于 MinMax 原则的改进 K-means 算法。

3.1 利用 MinMax 原则计算初始聚类中心

假设共有 n 个直观上可以划分成 k 类的聚类对象 x_i（i=1,2,3,…,n）。首先选出距离最远的两个对象记为x_{i_1}，x_{i_2}，以此为初始聚类中心的前两个点，其余的各点由递推公式表达。若已经选出 m 个聚类中心（m<k）,则第 m+1 个聚类中心应满足下式：

$$\min\left\{d\left(x_{i_{m+1}}, x_{i_r}\right), r = 1,2,\dots,m\right\} = \max\left\{\min\left[d\left(x_j, x_{i_r}\right), r = 1,2,\dots,m\right], j \neq i_1,\dots,i_j\right\}$$

公式（3-22）

其中$d\left(x_{i_{m+1}}, x_{i_r}\right)$表示第 m+1 个聚类中心到第 r 个聚类中心（r<m）的距离。

在 K 值已知的情况下，通过公式（2-11）的递推关系，即可求出 K 个初始聚类中心。为了更直观的阐述 MinMax 原则的思想，假设聚类对象的空间分布如图 3- 6 所示。算法首先找到聚类对象中距离最远的两个点 x_1，x_2，根据公式（3-22）的递推关系，第三个聚类中心应为距离前两个中心较近的所有其他点中距离最远的那个。因此在确定 x_1，x_2 后，下一个取出的初始聚类中心必定为 x_3 而非 x_3'。

根据上面的介绍，通过 MinMax 原则得到的初始聚类中心必定是十分分散的，考虑到聚类结果中不同的类要求尽可能彼此疏远，这些选出的初始聚类中心必然有一定的代表性，可以为 K-means 算法提供良好的指导。

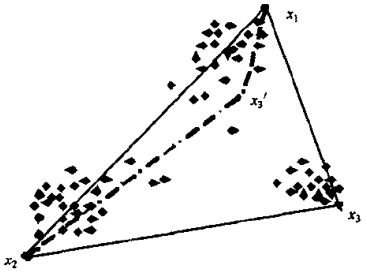


图 3- 6 MinMax 原则示意图

3.2 利用 MinMax 原则计算最优聚类数

根据 3.1，我们知道 MinMax 原则仅仅提供了一个递推关系，没有规定递推终止条件，亦没有提出对最优聚类数的估计方法。

参考 2.1 节的讨论，在此规定 MinMax 递推的截至条件是$k \le \sqrt{n}$，因为根据经验，n 个样本的聚类数一般不会超过$\sqrt{n}$，过为细化的聚类对数据挖掘中聚类分析下游的数据筛选和处理也造成麻烦。

3.1中在使用最小最大方法选择聚类点时，下一个聚类点的选择取决于距离

$$\min\{d(x_{i_{m+1}}, x_{i_r}), r = 1,2,\dots,m\} \qquad \text{公式（3-23）}$$

在此将这个距离定义为最小距离，经过分析，最小距离一般呈现出这样的规律：假设真实的类别数目为 K_T，在没有到达 K_T 之前，这个距离指标基本上表示类间的距离（较稀疏），而在超过 K_T 之后，这个距离则往往是类内的距离（较密集）。换而言之，通过 MinMax 原则选出的初始聚类中心之中，后一个中心与前一个的最小距离会在超过最佳聚类数之后产生一个突变，突变后的最小距离趋于平稳，因为其所代表的类内距离差别不大，这暗示着最小距离突变的位置往往就是最佳聚类数的取值。

图 3- 7 表示了一次聚类实验过程中最小距离的变化情况，可以看出当聚类数达到 5 类之后，最小距离趋于平稳。

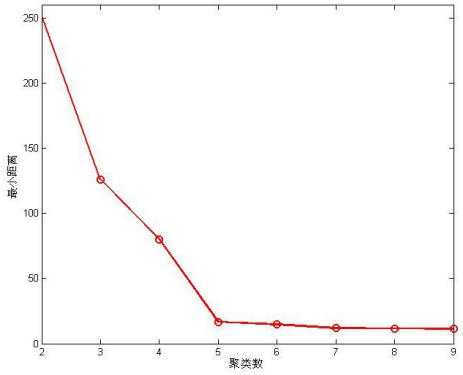


图 3- 7 最小距离随聚类数的变化曲线

为了准确找到最小距离突变对应的最优聚类数，本文通过两种不同的手段进行尝试。

3.2.1 利用深度函数查找突变点

设聚类数为 m 时对应的最小距离为 $mindist_m$，则聚类数取不同值时最小距离的变化量可表示为：

$$\Delta mindist(m) = mindist_m - mindist_{m+1} \qquad \text{公式（3-24）}$$

经过公式（2-13）的处理，度量类间和类内的最小距离的值被衰减，而突变点得到凸显，在公式（2-13）的基础上，定义深度函数 Dept(m)：

$$Dept(m) = \frac{\Delta mindist_m}{\Delta mindist_{m+1}} = \frac{mindist_m - mindist_{m+1}}{mindist_{m+1} - mindist_m} \qquad \text{公式（3-25）}$$

为了检验直接发查找突变点的效果，设计实验如下：

数据集来自五个具有不同期望的高斯分布，每个分布含有 30 个数据点，首先计算 150 个点两两之间的距离，获得前两个聚类中心，由$k \le \sqrt{n}$计算出最大聚类数为 12，然后利用公式（3-22）递推得到其余的 10 个聚类中心，进而根据公式（3-24）（3-25）求得 Δmindist(m)和 Dept(m)（如表 3- 2）。

通过图 3- 8 可以看出，利用深度函数可以较为准确地找到突变点，从而确定最优的聚类数，同时，应该注意到在 k 取 8 时同样出现了一个深度较大值，这是受到数据样本的具体空间分布影响的，虽然后面的最小距离衡量的是类内距离，但所选取的初始聚类中心之间的较小的距离差异仍有可能被公式（3-25）放大，因为这一步求的是最小距离差之间的比值，是一个相对大小，与绝对大小无关。因此直接法查找突变点虽然可以较为准确地放大突变点与突变前后最小距离的差距，但仍受到样本分布的影响，因此其鲁棒性有待提高。

m	mindistm	Δmindist(m)	Dept(m)
2	281.4394	151.1247	4.2770
3	130.3147	35.3339	2.5360
4	94.9808	13.9330	0.2135
5	81.0478	65.2676	57.3932
6	15.7802	1.1372	1.2499
7	14.6430	0.9098	0.4249
8	13.7332	2.1412	30.2857

9	11.5920	0.0707	0.1709
10	11.5213	0.4137	-1.8623
11	11.1076	0.3270	
12	10.7806		

表 3- 2 深度函数实验数据

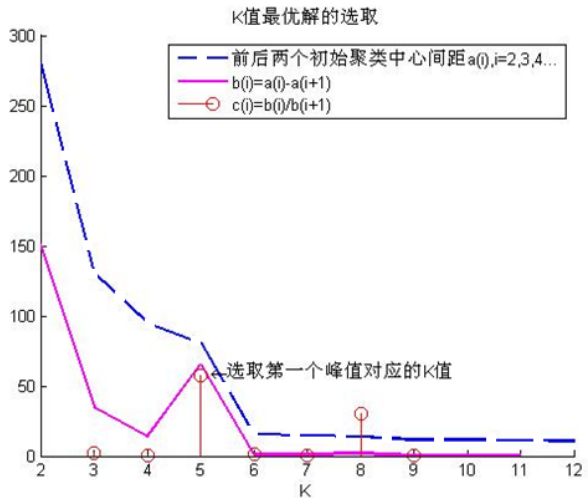


图 3- 8 利用深度函数选取 K 值最优解

3.2.2 阈值法查找突变点

根据 3.2，假设最优的聚类数为 k，当按照 MinMax 原则选取第 k+1 个点时，该点必然应该属于由最优聚类数划分出的某一个类，即该点与之前选出的某一个初始聚类中心之间的距离必然属于类内距离这一数量级，而远小于类际距离。如果能够找到类际距离与类间距离的分界点，记为阈值 T，即可通过判断下一聚点与之前选出各聚点的距离与 T 的关系来判断是否达到最优聚类数 K。

这里尝试引入全局中心 Cp(x,y)和全局尺度 T 的概念来作为划分类内距离和类际距离的阈值 T。另 n 个样本 $s_i(s_{xi},s_{yi})$ (i=1,2,…,n)的全局中心为：

$$Cp(x,y) = (Cp_x, Cp_y) \qquad \text{公式（3-26）}$$

$$Cp_x = \frac{\sum s_{x_i}}{n} \qquad \text{公式（3-27）}$$

$$Cp_y = \frac{\sum s_{y_i}}{n} \qquad \text{公式（3-28）}$$

则全局尺度 T 为：

$$T = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(S_{x_i} - Cp_x\right)^2 + \left(S_{y_i} - Cp_y\right)^2} \qquad \text{公式（3-29）}$$

根据上式，全局中心为所有样本的中心，全局尺度则是各个样本到这个中心的距离的均值。

全局尺度很含义在于为判断最小距离提供了一个绝对的参考值，相比于 3.2.1 中提到的方法，这个参考值不会受到样本空间分布随机性的影响，能够比较客观地对下一个初始聚类中心是否有效作出判断。

针对阈值法查找突变点，此处设计如下实验：

数据集来自四个具有不同期望的高斯分布（见表 3- 3），每个分布含有 50 个数据点。

组别	均值（E_x，E_y）	方差（σ^2_x，σ^2_y）
第一组	（0,0）	（210,210）
第二组	（-110,40）	（216,210）
第三组	（-110，-30）	（213,211）
第四组	（117,39）	（210,220）

表 3- 3 数据集详情

根据公式（2-18）可以计算出这 200 个样本的全局尺度 T=90.9775，考察任意两个初始聚类中心的距离（如表 3- 4），可以发现前四个初始聚类中心之间的距离都在 T 之上，而第五个初始聚类中心与第一个中心之间的距离为 76.51（图中标记阴影），说明第五个初始聚类中心应该归属于第一个初始聚类中心所划分出的那一类，据此，选取 k=4 为最优聚类数，与实际结果相一致。

	1	2	3	4	5	6	7	8	9	10	11	12
1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
2	301.70	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
3	156.26	153.59	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
4	244.77	117.61	146.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
5	76.51	263.27	109.69	237.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
6	217.19	84.52	73.50	92.46	181.85	0.00	0.00	0.00	0.00	0.00	0.00	0
7	262.68	66.48	106.67	144.31	212.50	63.77	0.00	0.00	0.00	0.00	0.00	0
8	258.54	62.63	128.62	56.52	233.50	57.17	91.90	0.00	0.00	0.00	0.00	0
9	57.51	244.59	99.41	195.84	56.41	160.08	205.42	203.68	0.00	0.00	0.00	0
10	155.64	146.78	45.77	108.06	131.00	62.91	119.03	104.68	99.49	0.00	0.00	0
11	118.21	198.53	44.95	181.53	64.75	117.68	149.23	171.02	65.41	73.51	0.00	0
12	261.41	41.46	112.13	105.74	221.81	44.82	41.04	51.33	204.08	107.73	157.07	0
13	283.53	83.59	165.45	47.02	266.38	96.62	129.14	40.38	231.29	135.48	205.93	88.1373

表 3- 4 初始聚类中心间最小距离

3.2.3 实验分析与比较

为了横向比较深度函数法和阈值法估算最优 k 值的效果，设计如下实验：

所采用的数据集中各分布的期望不变，方差分别选取 100 和 300，用来考量算法对不同样本离散度的适应性，每个分布中样本个数分别选取 10,15,20（样本总数即为 40,60,80），分别利用深度函数法和阈值法对上述样本进行 100 次 k 值估计，得到估计准确率如表 3- 5 所示。

高斯分布方差	σ^2=100			σ^2=300		
样本数	40	60	80	40	60	80
深度函数法准确率	65%	73%	80%	47%	40%	46%
阈值法准确率	68%	63%	60%	79%	76%	65%

表 3- 5 深度函数法与阈值法准确率比较

根据实验结果，深度函数法对分类较明显的样本的聚类个数能够给予较准确的估量，其估计准确率达到 70%左右，而随着样本中每一类的离散程度加大，算法的准确率受到很大的影响，相较而言，阈值法由于引入全局尺度，在区分类与类的问题上能够给出比较准确的度量，而且鲁棒性更好，但在实验中依然发现，当类间距离>>类内距离时，阈值法可能会在综合考虑全局尺度的基础上，将两个相距较近的类当做一个类，从而影响到估计的准确率，这也解释了为什么当 σ^2 较小时，阈值法估计的准确率会下降。

综合来看，相较于深度函数法和距离代价函数法，阈值法受样本空间分布的影响较小，估值准确率普遍达到 70%以上，是一种比较有效的解决 k 值无监督选择的方法。

第4节　一种基于顺序查询的改进方法

传统的层次聚类要求每一层都要重新计算距离矩阵，这样会大大的增加算法的时间复杂度，根据时间属性的特点，本节设计了一种基于顺序查询的时间属性聚类方法。

4.1 利用顺序查询法降低时间复杂度

根据时间属性有一定的次序性这一特征，本文尝试首先将初始数据按顺序进行排序，当第一次合并了其中最小的一对数之后，产生的新的节点也必然处于相应的位置，因此无需再次进行距离矩阵的计算，即可继续进行下一层的聚类，减少了算法的时间复杂度。

通过进一步实验发现，对于时间属性，我们只关心某几个临界值所对应的聚类情况，即日期层面（$ds<10^5$），月份层面（$ds<10^7$）以及年份层面（$ds<10^9$），所以传统层次聚类将任意两个相似度最近的聚类样本合为一类创建一个新层这种思想并不适用于我们的研究目的。

根据上述分析，在基于排序的基础上，引入查询机制，顺序检索每一个样本，将未超过指定阈值的样本合并为一类，超过指定阈值的样本则新建一类，检索结束后输出聚类结果。

4.2 算法实现

这里给出顺序查询法算法的伪代码：

1) 初始化排序终止标记 flag=1，类别标记 count=1 记样本总数为 n
2) while(flag==1)，do
3) for i=1:n-1

   if s(i+1)<s(i)，交换 s(i+1)和 s(i)，同时令 flag1=1；
4) end while
5) for i=1:n-1

   if (s(i+1)-s(i)>1000000) {count=count+1};

   s(i)属于第 count 类;
6) end for
7) 输出聚类统计

4.3 实验分析与比较

经过实验，顺序查询方法能够准确地将时间样本按照要求进行聚类。在第 3 章 1.3 中对 120 个数据样本进行聚类的实验中，利用 MATLAB 对传统聚类算法和改进后的顺序查询算法进行了时间统计，原算法运算时间为 0.022248s，改进后的算法运算时间为 0.000359s，算法速度提升 98.4%。说明算法在保证相同准确率的前提下，速度得到了大幅度的提升。

第5节　实验结果与分析

本节将针对本章提出的各种聚类算法及其改进，利用 Rand 值评价标准（见第 2 章 2.4）进行有效性评估。

实验样本来自包含五个具有不同期望的高斯分布的样本生成器，每个分布含有 50 个数据点，总计 250 个空间属性样本数据（见表 3- 6）。参与对比的算法有传统 K-means 算法，基于 MinMax 原则[1]改进的 K-means 算法，以及基于 MinMax 原则改进的 GMM 算法。

组别	均值（E_x，E_y）	方差（σ^2_x，σ^2_y）
第一组	（90,0）	（216,216）
第二组	（0,-40）	（216,216）
第三组	（-110,0）	（216,216）
第四组	（0,40）	（216,216）
第五组	（0,0）	（210,210）

表 3- 6 数据集详情

具体实验步骤：

1) 利用样本生成器生成一组数据点；
2) 分别利用三种算法进行聚类分析，其中传统 K-means 算法的聚类数给定为 5，其余算法自动计算聚类数；
3) 对各算法的聚类结果求 Rand 值；
4) 返回步骤 1，重复十次并计算 Rand 值的平均值

[1] 此处采用较优的阈值法选择聚类数，阈值法与深度函数法查找突变点的对比实验参见第 3 章 3.2.2

实验结果如表 3- 7。

Rand值	1	2	3	4	5	6	7	8	9	10	均值
传统Kmeans	0.799	0.7986	0.8006	0.7987	0.7973	0.7981	0.812	0.803	0.7964	0.7973	0.8001
改进Kmeans	0.8406	0.8554	0.8549	0.8455	0.8689	0.8118	0.8033	0.8448	0.8358	0.8451	0.84061
改进GMM	0.7872	0.8535	0.8319	0.8126	0.8342	0.8129	0.806	0.8561	0.8346	0.7742	0.82032

表 3- 7 改进前后算法 Rand 值比较

根据表中数据绘制直方图（图 3- 9），从图中可以直观的表现出改进后的算法的准确度更高，其中改进后的 K-means 算法的准确度最高，达到 84.061%，与传统算法相比提升了 5%，改进后的 GMM 算法准确度达到 82.032%，与传统算法相比提升了 2.5%。

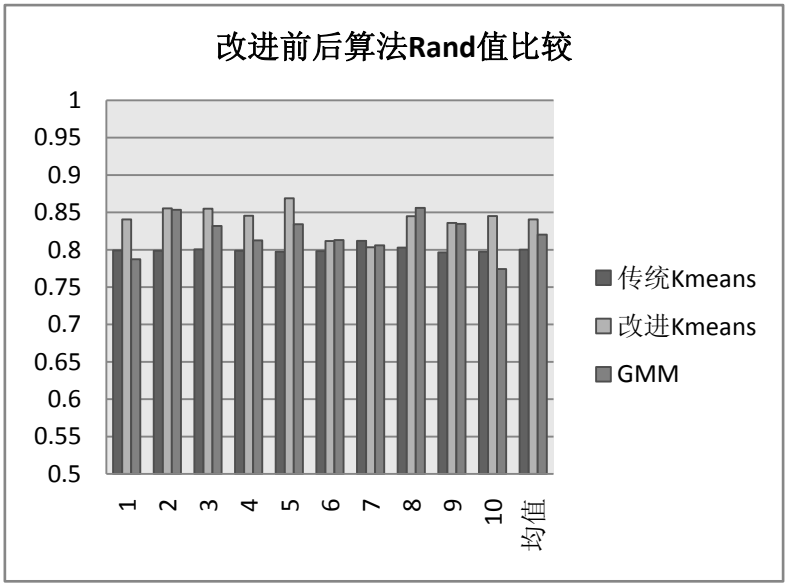


图 3- 9 改进前后算法 Rand 值比较

为了考察优化之后的算法在解决实际问题的能力，又进行如下实验：

聚类样本为从 3721 条国际新闻中提取出的新闻发生地的地理坐标，通过改进的聚类算法进行聚类分析，结果如图 3- 10 所示。图中红色点表示样本，绿色叉表示根据样本的分布进行聚类的结果，通过这一无监督的聚类过程，我们可以从中得到这 3721 条消息的空间分类情况：

1) 消息可以分为八类；

2) 北美洲三类：美国西海岸，美国中部，美国东海岸；

3) 欧亚大陆四类：欧洲，中亚地区，远东地区，东南亚及太平洋地区

4) 南半球一类：南极大陆。

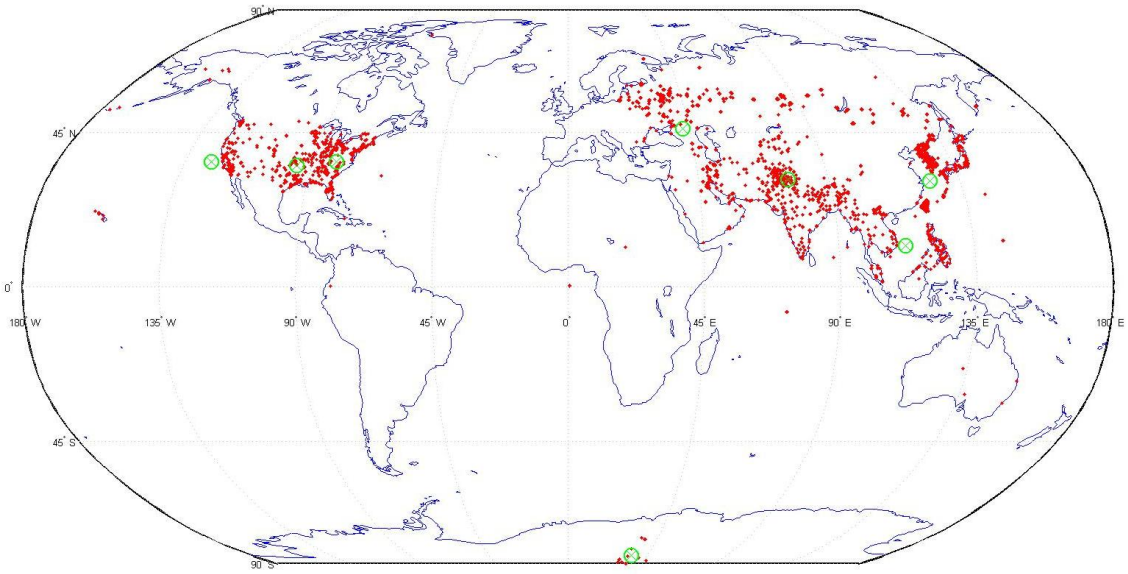


图 3- 10 全球地理数据实验

以上实验再次证明将聚类分析的思想引入地理信息系统，可以从庞杂而相互独立的消息中发现潜在的联系，提高了消息的可读性，降低了人工审阅、处理信息的难度，为进一步将消息表示服务。

第6节 本章小结

本章根据数据时空属性的特征，尝试通过不同算法对数据进行聚类研究，分别考察了基于划分的 K-means 算法，基于模型的高斯混合模型算法以及基于层次的聚类算法，通过对这些传统算法的分析，总结了他们各自的优势与不足。

首先传统的 K-means 算法与 GMM 算法虽然能够有效地进行聚类划分，但算法依赖于人工设定聚类数及单一高斯分布数，无法达到本文的研究目的，即自发的对海量数据进行聚类分析，因此本章的第 2 节和第 3 节对算法的改进中提出了三种自发选取聚类数的方法，实验证明这些算法能够相对准确地确定聚类数。

其次，上述算法在聚类过程中只能得到局部最优解，在传统聚类算法中初始聚类中心随机给定的情况下，很难得到稳定的全局最优解，从而严重影响到了聚类算法的准确率。考虑到上述问题，本章第 3 节利用 MinMax 原则计算基于全局的初始聚类中心，为聚类算法提供了较为准确的参考，实验证明基于 MinMax 原则改进的 K-means 算法和 GMM 算法的聚类准确率在一定程度上得到提高。

最后，考虑到基于层次的聚类算法的时间复杂度较高的问题，本章在第 4 节提出了一种基于顺序查询思想的改进算法，通过先排序，再进行阈值筛选的方法大幅度的提高了算法的运行速度。

本章第 5 节通过实验的方法综合讨论了各种改进算法与传统算法在聚类准确性上的差异，给出了客观的评价，实验表明在解决基于数据时空属性的聚类问题上，本文选择的聚类算法是准确而有效地，同时本章提出的改进算法进一步提高了原算法的准确性和鲁棒性，同时实现了从数据集到聚类集的自发组织。最后利用真实数据进行应用性实验，得到了较好的结果，印证了将聚类分析的思想引入地理信息系统的可行性。

第4章 针对数据时空属性的信息组织及软件实现

通过上一章的研究，传统的聚类算法针对本文的研究目的和研究对象得到了理论上的改进，面对工程上应用的需求，本章首先提出对于含有多维属性的数据的信息组织策略，进而通过.Net 平台初步设计实现一款聚类分析软件。

第1节　针对数据时空属性的信息组织策略

信息组织即信息的有序化与优质化，也就是利用一定的科学规则和方法，通过对信息外在特征和内容特征的表征和排序，实现无序信息流向有序信息流的转换，从而使信息集合达到科学组合实现有效流通，从而促进用户对信息的有效获取和利用。

本章提到的信息组织策略主要是指空间属性和时间属性的聚类组合以及对聚类结果的存储和表达。

1.1 多维度聚类存在的问题

对于含有多个维度的聚类对象，传统的聚类算法会使用同样的相似度评价标准来对待，比如在讨论含有空间属性的聚类算法中，欧氏距离被当做相似度的评价标准同时作用于 X 坐标与 Y 坐标，从而得到的是考察了 X、Y 坐标位置后的二维的聚类中心[21]。对于同维度且权重相当的多维数据属性，这种聚类方法是可行的[22]。

在实际应用中，如文本分析、染色体序列研究、商业中的客户偏好分析等都可以对多维度对象进行广义的欧氏距离计算，得到其分布情况。对于含有 m 维属性的据类对象 x_i 和 x_j，它们的广义欧氏距离表示为：

$$ds(i,j)=\sum_{k=1}^{m}\sqrt{|x_{i_k}-x_{j_k}|^2} \qquad \text{公式（4-1）}$$

而对于权重不同的属性，则可引入权重因子ω，对不同的维度进行加权，然后取广义的欧氏距离。令权重向量ω(k)表示第 k 维德权重，则广义加权欧氏距离表示为：

$$ds(i,j)=\sum_{k=1}^{m}\omega(k)\sqrt{|x_{i_k}-x_{j_k}|^2} \qquad \text{公式（4-2）}$$

以上讨论仅限于聚类对象的各个属性维度相同的情况，对于本文所讨论的数据对象，由于时间属性为一维属性，空间属性为二维属性，二者的相似度评价标

准亦不相同，无法直接利用公式（4-1）或（4-2）联合聚类。

从另一个角度来看，即便得出了同时考虑时间顺序与空间位置的聚类结果，这种结果也是无意义的。假使我们联合考察时空属性的到一个聚类结果，属于同一类的两条消息不仅仅可能是发生于相近时间和相邻地点（这些消息对是有价值的），同时也可能是发生于极相近时间但相去甚远的两个地点（这些消息并没有考察价值），因此这样的聚类结果无法满足聚类分析下游的信息处理需要。

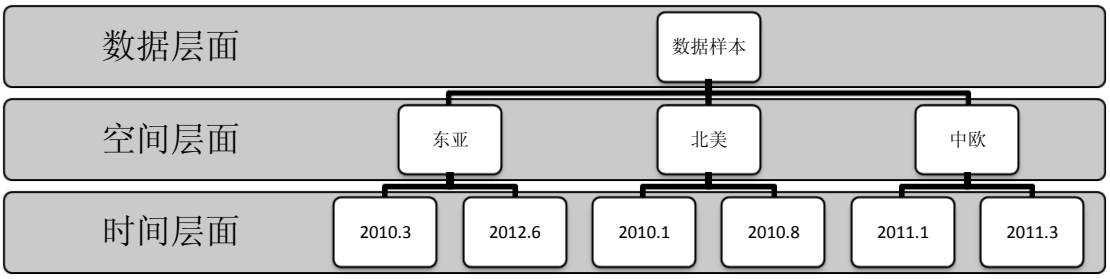
1.2 多维度聚类的层次化

作为聚类的观察者，我们可能首先更关心消息发生的地点分布情况，其次，关心在某一个地区内不同时间都产生了哪些消息；亦或是首先关心在不同的时间段内都发生了那些消息，再进一步想要知道同一时间发生的消息在地理分布上的特征。

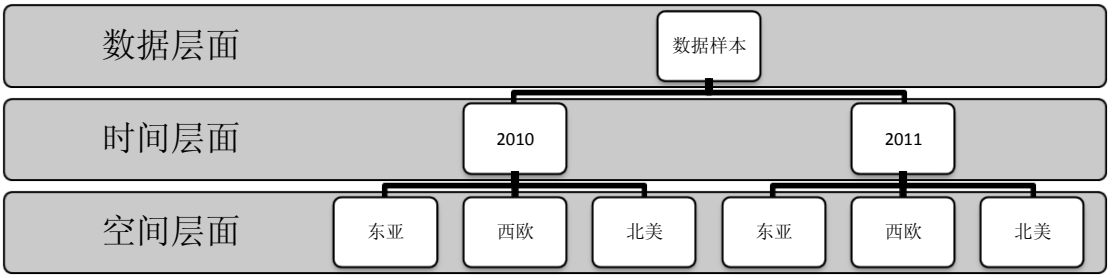
根据上面的描述，提出一种多维度聚类的层次化策略：

1) 根据聚类分析的目的以及用户的要求，首先对多维对象中的某一个属性进行聚类，其他属性进行保护处理；
2) 在第 1 步聚类结果的基础上，针对每一个结果的其他属性进行二次聚类，其聚类方法及相似性对量准则可以重新选取，并对无关属性进行保护处理。

下面用层次化的树状图分别举例表示了这种聚类的层次化思想的两种不同实现：



A．先空间后时间



B．先时间后空间

图 4- 1 多维度聚类的层次化过程

实验证明，这种层次化的数据组织方式更符合用户习惯，同时为聚类结果的表达提供了便利，附录 1 展示了一次对 120 个含有时空属性的样本进行联合聚类的结果，聚类采用先空间后时间的方式，首先得到了四个空间聚类中心(51.8372,-0.1823),(-51.8717,-0.0253),(0.3407,-52.0117),(0.3488,51.4100)，继而围绕每一个聚类中心，在该类中进行按月的时间聚类，并将结果分层表示出来。

第2节　聚类分析软件设计

为了能够直观地演示各种算法的聚类效果，也为了能够方便快速地应用各种算法解决实际问题，利用 Microsoft Visual Studio 2005 及 MATLAB 混合编程，初步设计了一个 Windows 程序，实现了包括数据导入，算法选择，结果表示在内一系列的功能，提高了研究的实用性。

2.1 MATLAB 与 C#混合编程

2.1.1　混合编程概述

MATLAB 和 C#作为本文研究的主要两个平台，分别承担了不同的工作：其中 MATLAB 作为一款具有强大矩阵运算、数据处理和图形显示功能的软件，拥有完善的函数库，用极少的代码即可实现复杂的运行，编程效率极高，在算法学习、实现和改进的过程中都起到了重要的作用，但如果利用其编写应用软件，其存在的问题就暴露出来了：由于 MATLAB 使用的是一种脚本语言，他的执行是逐行解释执行的，也就是边解释便执行，程序中的所有变量都是用 MxArray 来实现的，所以为了保证通用性，它的执行效率非常低，同时其生成的 M 脚本依赖于 MATLAB 这个应用程序环境，这就大大的制约了程序的可移植性和通用性。

相较而言，C#语言编写的程序执行效率高，内存利用率高，图形化编程简单便捷[23]。二者结合使用，可以取长补短，缩短开发时间[24]。

2.1.2 混合编程的实现

下面描述了 MATLAB 与 C#混合编程的具体实现过程：

1) 在 MATLAB 中利用 Deploytool 工具生成一个项目（project）[1]；
2) 在该项目下可以创建属于该项目下的类；
3) 添加写好的 M 文件到该类下，其中的 M 文件包含完整的函数，这个函数在 C#中会被当做一个方法来调用；
4) 运行 Build，生成.net 组件，同时会生成动态链接库（.dll 格式）；
5) 在 Visual Studio 中通过解决方案资源管理器引用这个动态链接库，然后通过调用下面类中的方法来调用部署的 M 文件中的函数。

2.2 功能设计

2.2.1 功能概述

该软件的主要功能包括：

1) 对本文讨论过的各种聚类算法进行聚类实验
2) 对含有时空属性的样本进行层次化联合聚类

其中第一项功能的实现依赖于三个主要模块：算法测试数据输入模块、核心算法模块以及数据输出模块。为了实现算法测试功能，其每一项模块都为用户提供了诸多选项，如数据输入模式可以选择人工输入数据，或使用数据生成器自动生成；核心算法可以指定 K 值以及不同算法。

而第二项功能的实现主要通过数据导入模块，核心算法模块以及数据输出模块组成。由于该功能主要目的是完成含有多维属性的聚类分析，故重点在于算法的效率及结果的表达，因此选用的核心算法是经过第三章实验分析后得出的最优算法，经过层次联合聚类，结果通过多层次列表的方式予以表达。

图 4- 2 描述了软件各模块的完整功能。

1 体现在 C#中就是一个 namespace

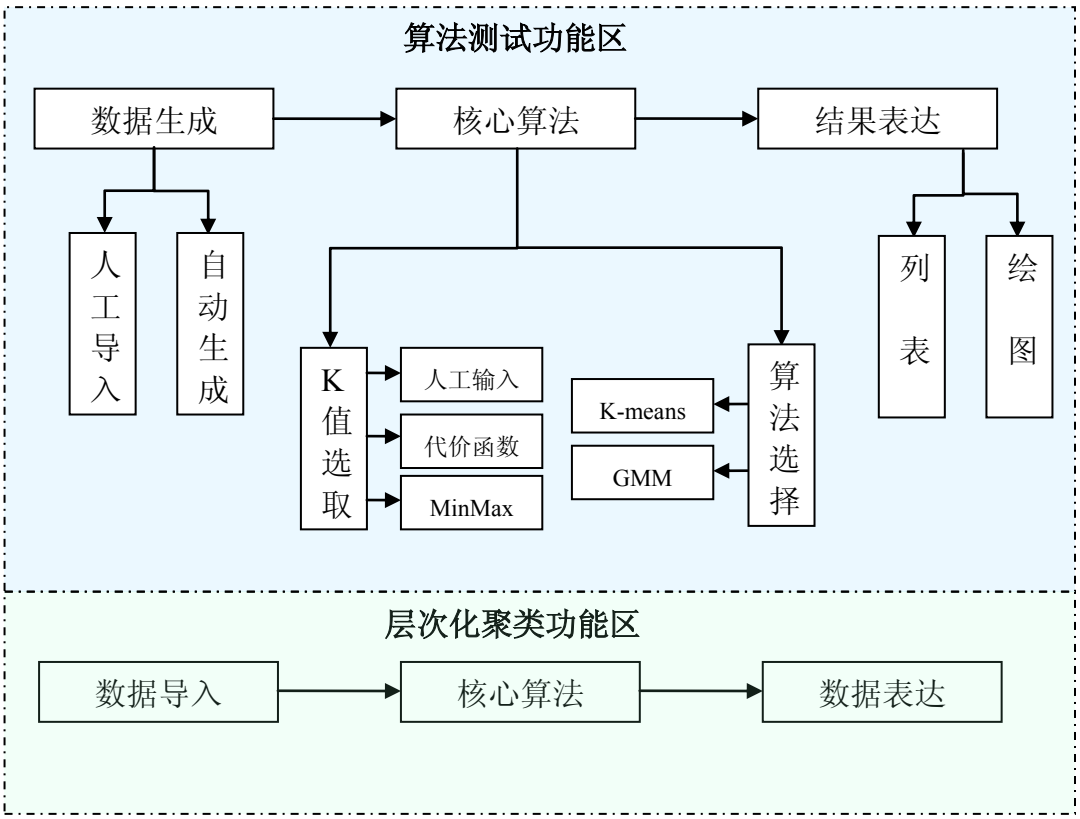


图 4- 2 模块示意图

2.2.2 算法测试数据输入模块：

1) 自动生成选项：能够自动生成 N 簇符合不同高斯分布的随机点。通过输入生成的分布数 n，每组个数 m，均值 μ，方差σ^2来控制生成数据的数量，位置，形状。同时，各簇数据的 μ 值可以作为评价聚类准确度的标准。
2) 人工导入选项：可以人工导入数据；

2.2.3 核心算法模块：

1) K 值选择选项：可以选择手工输入 K 值，适用于传统 K-means 算法和 GMM 算法；
2) 可以选择采取代价函数方法计算 K 值，适用于传统 K-means；
3) 可以选择采取最小最大原则求 K 值和聚类中心，适用于传统 K-means 和 GMM 算法。
4) 算法选择：提供 K-means，GMM 两种算法以供测试。

2.2.4 数据输出模块：

1) 空间数据输出：通过文本框和 GDI 绘图方式显示聚类结果；

2) 时空二维数据输出：通过文本框中按照一定层次列表输出。

2.3 界面设计

软件界面包括主界面，层次化聚类演示界面，以及其他弹出式窗口。

2.3.1 软件主界面

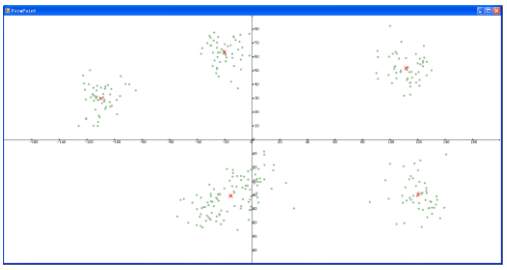
软件主界面按照功能模块划分为数据生成区、算法选择区、结果输出区以及按钮区。用户需要按照顺序依次完成各项操作。软件主界面截图见图 4- 3，各功能模块截图见图 4- 4。

图 4- 3 主界面截图

a.数据生成模块

b.核心算法模块

c.数据输出模块（列表）



d.数据输出模块（图像）

图 4- 4 功能模块截图

2.3.2 聚类演示界面

为了突出本软件的另一大功能，即对含有时空属性的数据的层次化聚类，将层次化聚类的功能在一个新的窗体中展示。其界面截图见图 4- 5

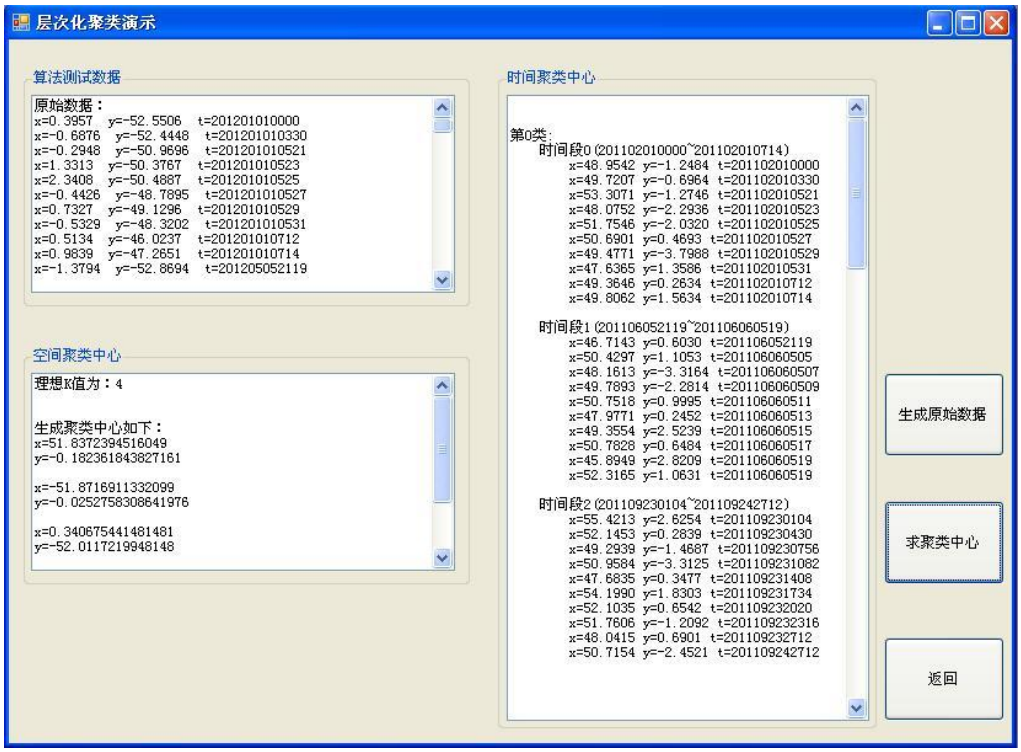


图 4- 5 层次化聚类界面截图

2.3.3 其他界面

本软件还为用户提供了样本生成器生成聚类样本的参考中心。样本生成器支持生成最多 10 类属于不同高斯分布的样本点，每类样本的方差默认设置为 100，期望已预先设置，并在软件中给出。见图 4- 6。

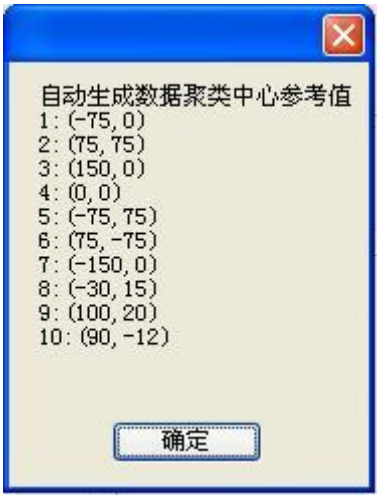


图 4- 6 聚类中心参考值截图

第3节 本章小结

本章首先讨论了含有多维属性的聚类样本的信息组织策略，根据本章 1.1 的讨论，传统的对与多维属性的聚类方法无法满足本文的研究需要：首先，传统的聚类算法会使用同样的相似度评价标准来对待每一个样本属性，这种思想无法解决对本文的研究对象，即含有不同维度的两个属性的聚类分析，其次，同时考虑时间顺序与空间位置的聚类结果不符合人类的认知习惯，因此无法满足聚类分析下游的信息处理需要。基于如上考量，本文提出了多维度聚类的层次化概念，实验证明，这种层次化的数据组织方式更符合用户习惯，同时为聚类结果的表达提供了便利。

基于本文的研究内容，本章的第 2 节设计了一款聚类分析软件，其功能在于对本文讨论过的各种聚类算法进行聚类实验并对含有时空属性的样本进行层次化联合聚类。该软件既是对第三章中所提到的各种算法及改进算法研究的总结，也是对本章第 1 节中提出的信息组织策略的实现，通过该软件的设计，本文所进行的各项理论研究的实用性得到了论证，为在地理信息处理系统中引入聚类分析策略提供了产品原型。

第5章 总结与展望

本章对全文的研究工作进行全面总结，并对后续研究工作进行展望。

第1节 论文总结

根据本文第一章对于问题提出背景的阐述，在面对海量消息的组织、处理、显示这一问题上，本文选择在地理信息系统表达消息对象的过程中引入聚类分析思想。聚类分析在 GIS 系统中的引入使系统能够根据现有的数据自发对其进行动态组织、分类，从而帮助用户发现数据潜在的规律。

针对基于数据时空属性的聚类研究，本文首先在第二章对时间属性、空间属性进行了相似度度量准则的建立，随后在第三章分别展开对时间、空间属性数据聚类算法的讨论。

根据第三章第一节的分析，传统的 K-means 算法和 GMM 算法都能够对二维的地理空间数据进行有效聚类，但二者存在一个共同的问题，即需要人为设定聚类个数，这显然是与我们实现数据自发组织这样的研究初衷相背离，因此本文在第三章的第二小节提出了一种基于代价函数的改进方法，代价函数算法是根据“一个好的聚类的评判标准应该包括（不仅限于）使类与类之间尽可能疏远，而类间各点尽可能的紧凑”而建立的，通过考察不同聚类数下由类内距离和类际距离构成的代价函数，选取最优的聚类数。根据实验发现，当数据较大时，类际距离与类内距离往往不属于同一个数量级，无法直接比较或者相加，因此本文又尝试将类际距离与类间距离归一化，得到归一化的距离代价函数，实验结果表明改进后的类际距离和类间距离的尺度得到统一，代价函数的特征相对明显，代价函数的最低点（即最优解）十分清晰，受数据影响较小。

传统的 K-means 算法和 GMM 算法存在的另一个问题在于二者的聚类方法都试图通过迭代得到一个局部最优解，而这一迭代过程依赖于一组初始聚类中心，传统算法中初始聚类中心的选择是随机的，这种随机性直接导致了算法的不稳定，据此，本文第三章的第三节提出了基于 MinMax 原则的改进方法，从而保证在初始聚类中心数不大于最佳聚类数的前提下，初始聚类中心之间一定是全局中相距较远的点，这就为传统的 K-means 算法和 GMM 算法的迭代提供了一个较为可靠的参照。在 MinMax 算法的研究过程中，我们发现最佳 K 值的选取与初始聚

类中心之间的距离有一定的关联，因此提出深度函数法和阈值法两种思路来计算最佳聚类数，实验证明利用 MinMax 算法计算最优聚类数的准确率要普遍高于代价函数法。

考虑到基于层次的聚类算法的时间复杂度较高的问题，在第三章第四节提出了一种基于顺序查询思想的改进算法，通过先排序，再进行阈值筛选的方法大幅度的提高了算法的运行速度。

提出基于多属性数据聚类的层次化思想，针对含有多维属性的聚类对象提出层次化聚类的策略，实验证明这种策略符合人类认知的自然规律，使聚类结果相较于传统使用相同相似度度量准则的聚类算法更容易被用户理解。

基于本文的研究内容，设计了一款聚类分析软件，并在本文第四章的第二节给予介绍。其功能在于对本文讨论过的各种聚类算法进行聚类实验并对含有时空属性的样本进行层次化联合聚类。该软件既是对第三章中所提到的各种算法及改进算法研究的总结，也是对本章第 1 节中提出的信息组织策略的实现，通过该软件的设计，本文所进行的各项理论研究的实用性得到了论证，为在地理信息处理系统中引入聚类分析策略提供了产品原型。

综上所述，本文在基于数据时空属性的聚类研究上建立了较为系统的研究思路，并针对传统算法，给出了有效的改进，使其更适用于工程上的实用需求。实验结果基本满足研究预期。

第2节　研究展望

本研究虽然取得了上述成果，但由于研究时间较短，个人能力不足等原因，尚有许多方面有待改进：

1) 对于数据中时间属性的表达格式，当前方法虽然简单直观，但占用大量内存，一种更为科学的数据存储格式有待进一步的研究；
2) 除了本文讨论的基于划分和基于模型的聚类方法，可以尝试利用基于网格的聚类方法对空间属性进行聚类，有研究表明基于网格的算法对脏数据的敏感度更低，同时节省大量时间复杂度；
3) MinMax 算法中计算两两样本之间的距离这一步骤的时间开销会随着样本数量的增加而成倍增加，因此限制了算法在处理大量数据时的效率，尚需进一步改进；

4) 研究过程中开发的软件尚为原型，仍存在很多细节需要进一步改进，可以尝试在编程的过程中将各种算法封装为一个类，进而得以简化；

5) 在对消息的时间、空间组织的基础上，可以进一步研究基于关键词的聚类方法， 从而实现按照消息内容聚类。

结 论

本文在基于数据时空属性的聚类研究上进行了较为系统的研究，针对传统基于划分的 K-means 算法、基于模型的 GMM 算法以及基于层次的聚类算法进行分析讨论，结合传统算法的不足及工程上的需要给出了有效的改进方法和建议。

实验结果表明改进的算法在准确率、稳定性以及时间复杂度上都有所提升；研究解决了算法不能自发选择聚类数这一问题，实现了聚类的无监督化；同时研究为含有时空属性的消息对象的聚类提出了现实可行的数据组织方法，原本海量的、无组织的消息经过聚合分类，成为了少量有代表性的时间节点或空间聚点，这些时间节点和空间聚点能够按照用户的偏好有层次地显示出来，该过程完成了信息的组织，大大的缩减了人工成本，完成了研究的预期目标。

致 谢

值次文付梓之际，特向完成毕业论文期间给予我莫大帮助和指导的老师，与我共同学习奋斗的同学以及多年来给予我诸多帮助的家人、老师及朋友们表示由衷的感谢。

感谢母校吉林大学四年的培育之恩，让我从一个不经世事的懵懂少年，一步一个脚印地成长起来。一年一岁，寒来暑往，在您的庇护下我不仅学到了丰富的知识，打下坚实的专业基础，更磨练了意志，塑造了人格。在电子所短短三个月的时光，让我对未来三年的生活充满向往，然而对母校的眷恋之情却与日俱增，转眼即将挥手作别，面对您的一草一木，一楼一宇，心中充满不舍与感激。“求实创新，励志图强”，我会把它铭记在心。

感谢我论文的指导教师——中国科学院电子学研究所付琨研究员和黄宇助理研究员，在我完成毕业论文期间对我的指导和帮助，你们的批评和指正让我领教到做学术的严谨之风，和你们的讨论让我初窥到科研的乐趣，感谢我的研究生导师吴一戎院士，和您的那次促膝长谈让我对自己未来读研的三年充满了期待。

感谢吉林大学张彤副院长对我的帮助和鼓励，感谢母校所有曾经传授过我知识的恩师和曾经帮助过我的朋友。

感谢中科院电子所卢葱葱处长、王永老师以及八室葛蕴萍老师对我的悉心照顾。

感谢八室的宋俊师哥、中国科技大学的董文强、陈丽勇、吴凡，这三个月来同你们的学习与探讨让我受益匪浅。

最后感谢我的父母和女友对我生活上的悉心照顾、支持和忍让。

四年的学习和生活得到了太多人的帮助和支持，在此对所有关心、帮助、支持过我的人，衷心的说一声“谢谢!”

参 考 文 献

[1] Edwin L. Shuman, Practical Journalism: a complete manual of the best newspaper method[M]. New York: D. Appleton and company, 1903

[2] 吴信才. 地理信息系统的基本技术与发展动态[J]. 地球科学, 1998,23(4):0.

[3] J. Han, M. Kamber, Data Mining: Concepts and Techniques[M], San Francisco: Academic Press, 2001

[4] 韩家炜, 孟小峰, 王静等. Web 挖掘研究[J]. 计算机研究与发展, 2001,38(4): 405~414.

[5] 陈湘涛,李明亮,陈玉娟等. 基于时间序列相似性聚类的应用研究综述[J].计算机工程与设计, 2010,31(3):577~581.

[6] 孙吉贵,刘杰,赵连宇等. 聚类算法研究[J]. 软件学报,2008,19(1):48~61. DOI: 10. 3724/ SP.J.1001.2008.00048.

[7] 戴晓燕,过仲阳,李勤奋等. 空间聚类的研究现状及其应用[J]. 上海地质,2003,(4):41~46.

[8] 汤效琴,戴汝源. 数据挖掘中聚类分析的技术方法[J]. 微计算机信息,2003(19):3~4

[9] MacQueen J. Some methods for classification and analysis of multivariable observation. in: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability. Berkeley: University of California Press,1967(1)．281~297

[10] E. M. Voorhees. Implementing Agglomerative Hierarchic Clustering Algorithms for Use in Document Retrieval. Inf. Proc. 1986(22)465~476

[11] JiaWei Han, Michelin Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2001. 279~333

[12] Pang Ning Tan, Michael Steinbach, Vipin Kumar. Introduction to Data Mining. Posts & Telecom Press, 2004. 132~212

[13] N Beckmann, H Kriegel, The R×tree: An efficient and robust access method for points and rectangles. 1990-ortal.acm.org

[14] 邵峰晶. 数据挖掘原理与算法. 北京:北京:中国水利水电出版社, 2003

[15] 段明秀. 层次聚类算法的研究及应用:[硕士学位论文], 湖南: 中南大学, 2009

[16] 秦钰,荆继武,向继等. 基于优化初始类中心点的 K-means 改进算法[J].中国科学院研究生院学报,2007,24(6):771~777.

[17] 袁方,周志勇,宋鑫等. 初始聚类中心优化的 k-means 算法[J]. 计算机工程,2007,33(3):65~66.

[18]Jeffrey D, Adrian E, Raftery. Model-based Gaussian and non-Gaussian Clustering[J]. Biometrics, 1993, Vol 49, No. 3

[19]杨善林,李永森,胡笑旋等.K-means 算法中的 k 值优化问题研究[J]. 系统工程理论与实践,2006,26(2):97~101

[20]Shi Zhongzhi．Knowledge Discovery[M]．Beijing: Tsinghua University Press, 2002．

[21]李新运,郑新奇,闫弘文等. 坐标与属性一体化的空间聚类方法研究[J]. 地理与地理信息科学,2004,20(2):38~40.

[22]陈小瑜,余明. 基于空间聚类分析的福建省各县市经济发展水平研究[J].热带地理,2007,27(4):343~347.

[23]K. Watson, C. Nagel. C#入门经典(第 5 版)[M],北京:清华大学出版社,2010

[24]王素立,高洁,孙新德. MATLAB 混合编程与工程应用[M]. 北京: 清华大学出版社, 2008

附 录 A

第 0 类	第 1 类	第 2 类	第 3 类
时间段 0(201102010000~201102010714)	**时间段 0(201203010004~201203010718)**	**时间段 0(201201010000~201201010714)**	**时间段 0(201203010000~201203010714)**
x=48.9542 y=-1.2484 t=201102010000	x=-49.8213 y=-1.4067 t=201203010004	x=0.3957 y=-52.5506 t=201201010000	x=-1.1024 y=49.8596 t=201203010000
x=49.7207 y=-0.6964 t=201102010330	x=-52.1209 y=-2.6919 t=201203010334	x=-0.6876 y=-52.4448 t=201201010330	x=-0.4041 y=45.4788 t=201203010330
x=53.3071 y=-1.2746 t=201102010521	x=-49.0799 y=0.5678 t=201203010525	x=-0.2948 y=-50.9696 t=201201010521	x=0.1025 y=47.8039 t=201203010521
x=48.0752 y=-2.2936 t=201102010523	x=-48.4862 y=-3.1946 t=201203010527	x=1.3313 y=-50.3767 t=201201010523	x=-0.1426 y=51.3696 t=201203010523
x=51.7546 y=-2.0320 t=201102010525	x=-48.0821 y=0.0466 t=201203010529	x=2.3408 y=-50.4887 t=201201010525	x=1.3670 y=49.8773 t=201203010525
x=50.6901 y=0.4693 t=201102010527	x=-51.5455 y=-1.2537 t=201203010531	x=-0.4426 y=-48.7895 t=201201010527	x=0.2444 y=47.4984 t=201203010527
x=49.4771 y=-3.7988 t=201102010529	x=-48.9952 y=4.8697 t=201203010533	x=0.7327 y=-49.1296 t=201201010529	x=4.0563 y=48.5994 t=201203010529
x=47.6365 y=1.3586 t=201102010531	x=-49.7750 y=2.5457 t=201203010535	x=-0.5329 y=-48.3202 t=201201010531	x=0.6977 y=50.5579 t=201203010531
x=49.3646 y=0.2634 t=201102010712	x=-48.1529 y=-5.5832 t=201203010716	x=0.5134 y=-46.0237 t=201201010712	x=4.0350 y=47.7795 t=201203010712
x=49.8062 y=1.5634 t=201102010714	x=-48.8011 y=0.9868 t=201203010718	x=0.9839 y=-47.2651 t=201201010714	x=-1.6169 y=52.1801 t=201203010714
时间段 1(201106052119~201106060519)	**时间段 1(201207052123~201207060523)**	**时间段 1(201205052119~201205060519)**	**时间段 1(201207052119~201207060519)**
x=46.7143 y=0.6030 t=201106052119	x=-47.9923 y=-3.1263 t=201207052123	x=-1.3794 y=-52.8694 t=201205052119	x=1.1774 y=48.5673 t=201207052119
x=50.4297 y=1.1053 t=201106060505	x=-50.2950 y=0.5703 t=201207060509	x=0.6146 y=-55.2077 t=201205060505	x=-0.5819 y=54.0447 t=201207060505
x=48.1613 y=-3.3164 t=201106060507	x=-50.3292 y=0.3676 t=201207060511	x=1.3441 y=-47.9832 t=201205060507	x=1.3420 y=47.5853 t=201207060507
x=49.7893 y=-2.2814 t=201106060509	x=-47.7466 y=1.6720 t=201207060513	x=0.2064 y=-54.1046 t=201205060509	x=1.3281 y=50.4454 t=201207060509
x=50.7518 y=0.9995 t=201106060511	x=-54.7486 y=0.6106 t=201207060515	x=3.8680 y=-49.8507 t=201205060511	x=-4.8881 y=46.5989 t=201207060511
x=47.9771 y=0.2452 t=201106060513	x=-51.1283 y=3.5247 t=201207060517	x=-1.3608 y=-49.9207 t=201205060513	x=-2.9674 y=48.3819 t=201207060513
x=49.3554 y=2.5239 t=201106060515	x=-52.8411 y=-1.0754 t=201207060519	x=-1.6481 y=-45.0199 t=201205060515	x=-3.2222 y=48.6735 t=201207060515
x=50.7828 y=0.6484 t=201106060517	x=-50.8555 y=0.7323 t=201207060521	x=-3.9128 y=-50.1548 t=201205060517	x=0.8986 y=50.8974 t=201207060517
x=45.8949 y=2.8209 t=201106060519	x=-48.5495 y=1.4864 t=201207060523	x=2.0359 y=-51.1344 t=201205060519	x=3.2875 y=52.1067 t=201207060519
x=52.3165 y=1.0631 t=201106060519	x=-48.1536 y=0.1905 t=201207060523	x=1.9389 y=-49.4727 t=201205060519	x=-0.7308 y=50.6719 t=201207060519
时间段 2(201109230104~201109242712)	**时间段 2(201210230108~201210242716)**	**时间段 2(201208230104~201208242712)**	**时间段 2(201210230104~201210242712)**
x=55.4213 y=2.6254 t=201109230104	x=-52.2695 y=1.9699 t=201210230108	x=-0.1786 y=-49.4504 t=201208230104	x=1.8164 y=49.1658 t=201210230104
x=52.1453 y=0.2839 t=201109230430	x=-51.0533 y=0.7227 t=201210230434	x=2.0091 y=-49.8434 t=201208230430	x=1.2199 y=51.8235 t=201210230430
x=49.2939 y=-1.4687 t=201109230756	x=-49.6936 y=-1.7534 t=201210230760	x=0.4108 y=-51.3608 t=201208230756	x=-2.3515 y=51.7864 t=201210230756
x=50.9584 y=-3.3125 t=201109231082	x=-50.6526 y=-4.0369 t=201210231086	x=0.6502 y=-52.7338 t=201208231082	x=0.8888 y=50.2688 t=201210231082
x=47.6835 y=0.3477 t=201109231408	x=-49.3251 y=4.1559 t=201210231412	x=0.2526 y=-49.2923 t=201208231408	x=-1.6813 y=51.2773 t=201210231408
x=54.1990 y=1.8303 t=201109231734	x=-49.1057 y=-1.3518 t=201210231738	x=0.9838 y=-53.0027 t=201208231734	x=3.3905 y=50.9230 t=201210231734
x=52.1035 y=0.6542 t=201109232020	x=-52.0795 y=0.2311 t=201210232024	x=0.2273 y=-52.3080 t=201208232020	x=-0.0728 y=47.7931 t=201210232020
x=51.7606 y=-1.2092 t=201109232316	x=-50.3954 y=1.2593 t=201210232320	x=6.2327 y=-47.0233 t=201208232316	x=3.6582 y=51.6984 t=201210232316
x=48.0415 y=0.6901 t=201109232712	x=-54.7675 y=0.2540 t=201210232716	x=-2.6087 y=-50.9367 t=201208232712	x=-0.9505 y=48.5305 t=201210232712
x=50.7154 y=-2.4521 t=201109242712	x=-47.4389 y=-2.0230 t=201210242716	x=-4.1463 y=-50.3138 t=201208242712	x=1.3180 y=48.6496 t=201210242712