分类号：O29　　单位代码：10110

学　　号：s20110085

谱聚类算法及其应用研究　中北大学

中 北 大 学

硕 士 学 位 论 文

谱聚类算法及其应用研究

硕士研究生　张亚平

指导教师　杨明

学科专业　应用数学

2014 年 5 月 15 日

图书分类号 **O29** **密级 非密**

UDC 注 1

硕 士 学 位 论 文

谱聚类算法及其应用研究

张亚平

指导教师（姓名、职称） **杨明 副教授**

申请学位级别 **理学硕士**

专业名称 **应用数学**

论文提交日期 **年 月 日**

论文答辩日期 **年 月 日**

学位授予日期 **年 月 日**

论文评阅人

答辩委员会主席

2014 年 **5** 月 **15** 日

原 创 性 声 明

本人郑重声明：所呈交的学位论文，是本人在指导教师的指导下，独立进行研究所取得的成果。除文中已经注明引用的内容外，本论文不包含其他个人或集体已经发表或撰写过的科研成果。对本文的研究作出重要贡献的个人和集体，均已在文中以明确方式标明。本声明的法律责任由本人承担。

论文作者签名：_____ 日期：_____

关于学位论文使用权的说明

本人完全了解中北大学有关保管、使用学位论文的规定，其中包括：①学校有权保管、并向有关部门送交学位论文的原件与复印件；②学校可以采用影印、缩印或其它复制手段复制并保存学位论文；③学校可允许学位论文被查阅或借阅；④学校可以学术交流为目的，复制赠送和交换学位论文；⑤学校可以公布学位论文的全部或部分内容（保密学位论文在解密后遵守此规定）。

签　　名：_____ 日期：_____

导师签名：_____ 日期：_____

谱聚类算法及其应用研究

摘要

聚类分析是一种研究样本分类的统计方法，也是一种数据挖掘的方法，可以有效地实现数据结构的探测，在国际上已成近年机器学习领域的研究热点。谱聚类算法作为聚类算法的一个分支是一个重要的研究方向，以图论作为理论基础，基本思想是将聚类问题转化为图论中的对无向加权图的最优分割问题。与现有的其他典型的聚类分析算法相比较而言，谱聚类算法在聚类的过程中降低了对样本空间形状的要求，同时谱聚类算法还可以有效的克服一些经典聚类算法易收敛于局部最优解的缺点，可以得到收敛于全局的最优解。

本文在对前人研究谱聚类算法所取得的现有成果进行了系统的学习，对已有的相关算法进行了部分改进。具体的工作内容概括如下：

1) 首先介绍了关于聚类算法特别是谱聚类算法的基本概念以及理论基础，分析了谱聚类算法中的一些相关技术的已有的研究成果以及应用的现状，然后对谱聚类研究领域中所涉及的几个关键性问题进行了总结，归纳了谱聚类算法未来的几个关键的研究方向。
2) 针对传统的谱聚类算法中的两个基本的难点：相似矩阵的构造以及自动确定类的数目问题，本文通过定义的一种新的距离度量—密度敏感的距离和引入的特征间隙两个概念，提出了一种基于密度敏感的自适应谱聚类算法。新提出算法的有效性在模拟数据集以及 UCI 数据集上的实验中都得到了较好的验证，同时本文还计算了该算法与传统的 SC 算法的分类正确率，比较结果显示新算法的聚类性明显优于 SC 算法。
3) 针对 IPCM 算法对稀疏程度不同的样本集的聚类效果不理想，而且需要人工手动输入聚类数目的缺点，本文通过引入密度敏感的距离与特征间隙两个概念，提出一种基于谱聚类的自适应 IPCM 算法，该算法用密度敏感的距离代替传统的欧氏距离，并通过特征间隙的性质准确地计算出聚类数目。通过数据实验，证明了改进后的算法的有效性以及正确率都有所提高，同时该算法能够有效的弥补 IPCM 算法及谱聚类算法在各自单独聚类时所存在的缺点。
4) 将本文所提的密度敏感相似性度量引入到判别割算法中，代替原有的欧

氏距离，以此对判别割算法进行改进，在此基础上提出了一种基于密度敏感的判别割的图像单阈值分割算法。该方法在算法实现的复杂度和实现时所需存储空间等问题上都有了一定的改进。

关键词：谱聚类，密度敏感，自适应，图像分割，IPCM，判别割

Spectral clustering algorithm and its application research

Abstract

Cluster analysis is a statistical method for classification of the sample, but also a data mining method that can effectively achieve the detection data structures, machine learning has become a hot topic in recent years in the international arena. Spectral clustering algorithm is an important research direction in the clustering algorithm, its theoretical foundation from graph theory that changes the clustering problem into a graph theory problem without dividing the weighted graph. Compared with the other existing clustering algorithm, spectral clustering algorithm can be implemented any clustering problem in any sample space, and can overcome the phenomenon that the appearance of local optimal solution, at last obtained the optimal solution.

Based on the systematic study of spectral clustering algorithm this paper improves the part of the relevant algorithm. Specific content of the work are summarized below:

1) This thesis introduces the basis of knowledge of clustering algorithms and spectral clustering algorithm, and analysis the technology research and application of spectral clustering, summed up several key issues involved in the field of spectral clustering and several research directions of spectral clustering algorithm.

2) For the traditional spectral clustering of two key issues - how to define the similarity matrix and determines the number of classes automatically, this article put forward the algorithm that a kind of density sensitive adaptive spectral clustering algorithm based on the two concepts that density –sensitive distance and feature gap. The experiments of the new algorithm is effective on artificial datasets and UCI datasets, while the algorithm compares with the traditional algorithm SC in the classification accuracy, the comparison results show the new algorithm has better clustering effect.

3) For IPCM algorithm doesn’t have ideal clustering effect for different degrees of sparse data set, and you need to initialize the number of clustering, this article put forward the

algorithm that a kind of adaptive IPCM algorithm based on spectral clustering based on the two concepts that density –sensitive distance and feature gap. New algorithm use the density sensitive distance instead of Euclidean distance and calculate the number of clusters by the feature space accurately. The experiment proves the effectiveness of the improved algorithm, the algorithm can compensate for the lack of IPCM spectral clustering algorithm and weaknesses when each individual clusters that exist.

4) This article improve the D-cut algorithm by using density sensitive similarity measure mentioned in place of the original Euclidean distance, and put forward a new algorithm that a kind of image threshold segmentation based on density cut sensitive discriminated method. The algorithm use gray-scale weight matrix (instead of pixel-level image-based weight matrix) to describe the relationship between the image pixels, this method is less complexity and need less storage space than other graph-based image segmentation method.

Keywords: spectral clustering, density sensitive, adaptive, image segmentation, IPCM, discriminate cut

目　录

1 绪论

1.1 引言

聚类分析是一种研究样本分类的统计方法，也是一种数据挖掘的方法，可以有效地实现事物之间内在联系的探索。聚类算法的根本目的在于在一定的标准下将所给的样本自动分为相应的类。手工对数据进行分类在实际应用中具有很大的局限性 [1,2]。聚类属于无监督分类范畴，分类过程中根据所给分类对象自身所具有的某些特征区分其相互之间的相似程度。聚类算法已经越来越受到研究者的重视并且目前在许多领域都得到了广泛的应用，如：文本/语音识别领域的应用；数据挖掘以及异类数据分析领域的应用；以及机器学习中的图像分割与机器视觉的应用；此外，还在商业分析领域、市场营销分析领域、生物学领域、地理学领域以及心理学分析等诸多领域应用[3,4]。

谱聚类算法是聚类算法中的一个比较新的研究方向，其利用的是数据集的相似性矩阵中特征向量的性质对数据集进行聚类。谱聚类算法的思想是利用数据点之间相似性大小进行数据集聚类，此方法同样可应用于非测度空间中的聚类分析问题[5]。应用于谱聚类算法的一些常用准则也可以在一个新的空间中进行解释 [6]。

1.2 谱聚类算法

1.2.1 算法概述

谱聚类算法的理论基础来源于图论，其目的是把聚类转化为图论中的图分割。即假定将样本数据中的每个数据点视为一个图中的相对应的顶点 V，并且设定样本数据中的数据对之间都有着一定的相似性，此相似性用两点间边 E 的权值表示，此时便得到了一个无向加权图 G=(V，E)。基于图论的最优划分准则便是，使得在最终的划分结果中得到的子图内部的数据点的相似度最大，而属于不同的子图的数据点间的相似度则最小[7]。

图论中包含许多划分准则，根据这些准则可以概括谱聚类算法的实现方法为 [8-10]：

1) 定义计算数据点之间的权值的相似性的度量，构建可以描述数据点相似性的的相似矩阵；

2) 计算相似性矩阵的前 k 个最大的特征值所对应的特征向量，并且用这些特征向量构建新的数据特征空间，然后按照一定的划分准则对新的数据空间进行划分，具体划分准则如下：

其一，对于 2-way 的划分，即将样本点映射至一维空间（k=1）之中；其二，对于 k-way 的划分，即将样本点映射至 k 维空间（假定该空间具有 k 个正交的特征向量）之中。

3) 将原样本数据点集与由 k 维子空间 X 的行所构成的新数据点集进行一一对应，并对新的数据点进行聚类，聚类过程相应地也分为以下两种情况：

其一，2-way 划分，即依据目标函数的最优化原则，在一维空间中进行划分，然后在划分好的子图上进行迭代过程；其二，k-way 划分，即利用 k-mean、FCM、c-mean 等经典算法对新数据点集进行聚类。

依据上述给定的划分准则，谱聚类算法大致上可以分为迭代谱聚类法和多路谱聚类算法两类，其代表的算法分别为 SM 法和 NJW 法。

1.2.2 算法优点

相较于其他聚类方法，谱聚类算法有着较多的优势，概括如下[11,12]：

1） 算法思想相对简单且易于实现，在聚类的过程中，无需对数据的全局结构进行假设；

2） 在一定条件下谱聚类算法可以不受数据点维数的限制，只与数据集中的点的数目相关，因此对由于特征向量维数过高而引起的奇异值问题，在谱聚类算法中也可以得到有效的解决；

3） 可以有效的避免在聚类过程中局部最优解的出现，从而获得全局最优解。

因为谱聚类具有上述的诸多优点，所以在实际的工程中应用广泛，如语音识别领域、图像分割领域、视频分割领域、文本挖掘领域等。尽管谱聚类方法在前期的应用中已经取得了一定的成果，但目前该技术仍不够完善，算法本身仍存在着诸多问题[13]。

1.2.3 研究现状

近年来，基于图论的算法的研究与应用已成为学术界的一个热门。谱聚类算法由于在图论理论的基础上，因此也得到了较快的发展，谱聚类算法是一种基于图论的分割问题。该算法将数据点映射为带权的无向图，把样本数据点对应于图的节点，边的权重则代表样本点特征之间的相似性，然后利用某种分割准则得到图的最佳划分结果[14]。下面就从经典谱聚类算法的发展以及谱聚类算法最新进展两个角度，对谱聚类算法的研究现状做简要的介绍。

1) 经典谱聚类算法

a) 二路谱聚类算法

学者 Peron 与 Freeman 共同提出了 PF 算法，该算法的思想是利用相似矩阵的第一大特征值对应的特征向量进行聚类。对应于特征向量中零值的点划分为同一类，其余的点则划分为另一类[15]。之后学者 Shi 和 Malik 提出了 SM 算法，该算法认为相似矩阵 W 的第二小特征值所对应的特征向量（即 Fiedler 向量）比最大特征值所对应的特征向量包含了更多的有利于图划分的信息，在 Fiedler 向量中求可以使 N-cut(A,B)的值最小的划分点，然后将这一最小值同 Fiedler 向量中的值做比较，在 Fiedler 向量中大于该值所对应的点应划分为一类，其余的点应则划分为另一类[16]。Scott 和 Languet Higgins 提出了 SLH 算法，SLH 重定位算法在事先指定 k 值的基础上计算相似度矩阵 W 的前 k 个特征向量[17]。学者 Weiss 对 SM 算法与 SLH 算法进行了改进，并提出了 KVV 算法，与 SM 算法不同的是该算法是在 Fiedler 向量中寻找使 R-cut(A,B)值最小的划分点 i，尽管在实际的聚类应用中 KVV 算法依旧存在运行速度慢的缺点，但其他算法在分类过程中所存在的过分割问题在 KVV 算法中得到了有效的解决[18]。

b) 多路谱聚类算法

李小斌等人在矩阵扰动理论的基础上研究了谱及特征向量与聚类结果在理想情况下、分块情况下以及一般情况下的关系：如果顶点集合 V 包含 k 个彼此分离度很高的类，这时类的数目即为相似度矩阵 W 中那些所有大于 1 的特征值的总数。将相似度矩阵 W 的前 k 个单位正交特征向量组成一个新的矩阵 X，当两个顶点不属于同一类时，二者对

应于矩阵 X 中的行向量近乎于正交，反之二者对应的行向量则近乎于平行。并且在理想的条件下，当行向量间相互平行时，顶点集中的两个顶点则为同一类，而当行向量间相互正交时，顶点集合中的两个点则归于不同的类中[19]。NG 及 Jordan 等人提出了 NJW 算法，算法用拉氏（Laplace）矩阵的前 k 个最大特征值所对应的特征向量构造新的向量空间，新的空间中的点与原始数据集的点是一一对应的关系，之后再通过一些经典算法对样本数据集进行聚类[20]。

2) 谱聚类算法的新进展

Meila 提出了 MS 算法，该算法重新构造了相似度矩阵，并且引入了基于马尔可夫链随机游动过程的概率转移矩阵。MS 算法在图像分割中的应用已取得了较好的效果，但是 MS 算法的前提条件是度矩阵 D 中对角线上的元素值的差别不大，否则最终的聚类效果仍然不够理想[21]。

Zha 等人研究了一种基于二分图新的谱聚类算法，并且得出了二分图关联边权重矩阵的奇异值分解问题与最小化目标函数相对等的结论[22]。Dhillon 把两个经典算法 k-均值算法与 N-cut 算法相结合，给出了新的加权核 k-means 算法[23]。Xing 与 Jordan 共同对 N-cut 的半正定规划模型进行了研究，并发现 N-cut 本身无法获得最优聚类结果，但其可以在其他离散方法的基础上进行聚类，进而获得较好的效果[24]。

在有约束半监督学习中，王玲、焦李成等人提出在聚类搜索的过程中若有效利用先验信息可以得到更好的聚类结果 [25]。Bach 和 Jordan 重新定义了一个目标函数，引入随机游动模型，提出了新的聚类算法[26]。

1.2.4 关键技术

虽然谱聚类算法在以往的研究中已经有较大的进步，而且与其他一些聚类方法相比，其具有诸多的优点，但是谱聚类算法仍旧存在着诸多的不足，综合相关资料总结谱聚类算法需要研究的关键技术如下[27-30]：

1) 相似矩阵 W 的构造

通常情况下，相似矩阵的选择不仅受尺度参数的影响，而且受相似函数定义的限制。在大多数文献中，相似矩阵的计算公式为：$W_{ij}=\exp\left(-d^2\left(v_i,v_j\right)/2\sigma^2\right)$，式中的$\sigma$为

人工选取，算法能否成功关键在于尺度参数σ选择的合适与否。现有的谱聚类算法中σ的值一般全局统一，并都通过重复试验的方法来选取σ，受限于人为及时间因素。在现有的大多数资料文献之中，相似矩阵的构造多依赖于相应领域的先验知识，没有一个可以通用的规则，谱聚类中相似矩阵的构造需要进行系统的研究。

2) 聚类数目的自动确定

聚类数目能否有效正确地确定，将直接影响聚类的最终结果，现有的谱聚类算法都没有提出一个通用的自适应算法。

3) 特征向量的处理

特征向量的选取准则、怎样计算和使用等问题，在现有的理论研究中均还没有得到较好的证明。

4) 拉普拉斯矩阵的选取

谱聚类算法中有三种形式的拉普拉斯矩阵可供选择使用，但如何在不同的聚类环境中选择恰当形式的拉普拉斯矩阵才能得到较好的效果，该问题还没有得到较好的解决。如何根据实际的聚类问题选取相应的的拉氏矩阵，依旧需要进行大量的研究。

5) 海量数据中的应用

矩阵的特征值和特征向量的计算是谱聚类算法中的必要步骤之一，因此必须考虑计算时间对整个算法所带来的影响，在求解非稀疏矩阵的所有特征向量时，标准的解法需要$O\left(n^3\right)$，在大规模的计算和应用中，其计算过程的复杂度很大，如何将谱聚类算法高效地应用在海量数据的分类计算之中，也将是未来的一个研究方向。

1.3 本文研究的主要内容

本文在对谱聚类算法进行系统研究的基础上，对一些聚类算法进行了结合和部分的改进。全文的总体架构如下：

第一章：绪论。简单地介绍了谱聚类算法及其相关技术的研究以及发展现状，之后总结了算法中所涉及的几个关键技术。

第二章：谱聚类算法基础。对聚类算法及谱聚类分析算法的有关基础知识进行系统的介绍，其中包括图的基本知识、度矩阵、拉普拉斯矩阵、图的划分准则以及一些经典

的谱聚类算法等。

第三章：为了解决谱聚类算法中相似矩阵 W（W 能够有效反映数据点间的近似关系）的构建和聚类数目如何确定这两个关键问题，本章定义了密度敏感的相似性度量，引入特征间隙的概念。提出一种基于密度敏感的自适应谱聚类算法，并对算法结果进行了理论分析和实验验证。

第四章：将基于密度敏感的自适应谱聚类算法的思想引入到 IPCM 算法中，克服 IPCM 算法在处理疏密程度不同及高维数据集时出现的聚类问题，从而解决了 IPCM 单独聚类时的缺陷，并在人工数据以及 UCI 数据集上对算法进行了实验验证，并对算法结果进行了分析。

第五章：将密度敏感的相似性度量引入到判别割算法中，对判别割算法进行改进，提出一种基于密度敏感和判别割的图像阈值分割法，对算法进行了理论分析和实验验证。

第六章：总结与展望。对全文进行了回顾，分析了本人在谱聚类算法研究的过程中所存在的不足，并对在后续的研究工作中所需要完善的内容进行了总结。

2 谱聚类算法基础

本章主要介绍本文研究过程中所包含的一些基础知识，包括：常见的聚类算法及其分类、图以及图的矩阵表示形式、图划分准则以及一些经典的谱聚类算法等知识。

2.1 聚类分析方法概述

在聚类领域的研究中，已经有许多经典的算法。聚类是依照给定对象的相关属性，将对象聚集成类，使相似度高的对象尽可能的划分在同一类中，而相似度低的对象则划分到不同类中。在一些相关领域中聚类分析常作为一种预处理方法，为数据处理奠定基础[31]。由于聚类分析具有非常广泛的应用，因此许多学者在各种文献中，根据的聚类原理的差异，给出了为数众多的聚类算法。

1) 划分聚类方法

算法思想：假设要将给定的样本集划分为 k$(k \geq 1)$个类，并且规定在最终的分类结果中，每个类至少包含一个样本数据点，同时每一个样本数据点只能属于一个类。当类个数 k 确定时，算法将进行初始化划分，之后再经过迭代来对划分进行反复地改变，迭代的过程也是划分方案的效果也将不断提高的过程。一般情况利用目标函数来对划分标准进行衡量，算法在不断的迭代过程中目标函数也在不断的优化，所以划分聚类的方法又有另一种称谓——最优化方法。该方法的典型算法代表有：CLARA 法、k-means 法、CLARANS 法、PAM 法等[32-35]。

2) 基于网格的聚类方法

该方法是将对象空间进行量化，形成网络结构（该网格结构具有多分辨率的特征），将数据集的聚类问题转化为网络结构上的聚类问题。网格的聚类方法具有聚类速度快的优点，聚类所用的时间仅与网格结构的数目有关，无关乎数据集对象的数目。但是，该方法无法实现斜边界的检测，仅对水平或垂直的边界有效[36,37]。

3) 层次聚类方法

这个方法具体的可以分为两大类，第一，分裂型层次聚类方法，即在聚类开始时假定所有数据点都属于同一个类，在迭代的过程中，利用相应的准则一个类被一直的分解，

使之成为更小类，直到满足预定的条件时，分裂的过程停止。第二，合并型层次聚类方法，相反的，设每个样本点都是一个独立的类，之后把相似度高的类逐渐合并，当满足算法的终止条件时，停止迭代[38,39]。

4) 基于模型的聚类方法

该方法主要包括神经网络法以及统计学方法。其中统计学方法中最为典型的算法为COBWEB 算法，神经网络方法的代表方法有自组织映射算法 [40-41]。

5) 基于密度的聚类方法

主要思想：设定阈值，如果给定的一个区域中的数据点的密度值大于这个阈值时，便将这一区域其归类到与之相近的类里，该方法不仅能聚类圆形数据集，对其他任意形状的数据集均可以进行聚类，并且可以过滤数据集中的噪声点和孤立点[42]。

6) 基于图论的聚类方法

将给定样本集视为一个图的顶点集合，顶点间的边的权值表示记为样本点间的相似性大小，在这一假设下就将聚类问题转换成为了图的划分问题。常见的典型算法有CHAMELEON 法、谱聚类法以及最小生成树法[43-45]。

2.2 谱聚类算法的图论的基础介绍

2.2.1 图与图的表示

1) 基本表示[46-47]

设 G 表示一个图，且 G=（V,E），V 是图的顶点集合，E 是顶点间边的集合，边的个数用 m(G)=|E|表示，顶点的个数用 n(G)=|V|表示，分别简记为 m 和 n。如果两个边 $e_1,e_2 \in E(G)$ 含有公同的顶点，则视这两个边 e_1 和 e_2 相邻。

记 d(v)或 d_v 为顶点的度，即在图中以该顶点为端点的所有的边的数目。使用 b_{ij} 表示顶点 v_i 与边 e_j 的关联次数，称矩阵 $B(G)=(b_{ij})_{p\times q}$ 为图 G 的关联矩阵。使用 a_{ij} 表示图 G 中顶点 v_i 与顶点 v_j 之间的边，则称矩阵 $A(G)=(a_{ij})_{p\times p}$ 为图 G 的邻接矩阵。

设顶点 v_i 的度为 $d(v_i)$，则图 G 的度矩阵定义为 $diag\left(d(v_1),d(v_2),...,d(v_p)\right)$。

2) 图的矩阵表示[48-50]

以下给出了几种常见的关于图的矩阵表示形式：

a) 二值邻接矩阵：设图为$G_k=(V_k,E_k)$，其中V_k为点集，$E_k=V_k\times V_k$为边集，v_i，v_j为图中的点，i、j分别表示矩阵的行及列，则二值邻接矩阵的表达式为公式（2.1）：

$$A_k(i,j)=\begin{cases}1 & (v_i,v_j)\in V_k\\ 0 & otherwise\end{cases} \tag{2.1}$$

b) 加权邻接矩阵：在图G_k中，如果两点之间的距离表示为d(i.j)，则加权邻接矩阵的表达式如公式（2.2）所示：

$$A_k(i,j)=\begin{cases}W\cdot \mathrm{exp}\left(\dfrac{-d(v_i,v_j)^2}{\sigma^2}\right) & (v_i\ v_j)\in V\\ 0 & otherwise\end{cases} \tag{2.2}$$

2.2.2 相似矩阵、度矩阵及拉普拉斯矩阵

相似矩阵定义如公式（2.3）所示：

$$A_{ij}=\exp\left(-\frac{d^2(s_i,s_j)}{2\sigma^2}\right) \tag{2.3}$$

s_i表示第 i 个数据样本点，$d(s_i,s_j)$是数据点s_i，s_j之间的距离，通常取欧氏距离$\|s_i-s_j\|$，σ为尺度参数。

数据点的度可以有效的反应其周边其他数据的一些分布情况，度矩阵是由全部度值为对角元素所构成的对角矩阵，表达式如公式（2.4）所示：

$$D_{ii}=\sum_j A_{ij} \tag{2.4}$$

拉普拉斯矩阵以相似矩阵为基础生成，通常分非规范和规范两类。

非规范拉普拉斯矩阵的表达式为公式（2.5）：

$$L=D-A \tag{2.5}$$

规范拉普拉斯矩阵有两种形式，分别如公式（2.6）、公式（2.7）所示：

$$L_{nor1} = D^{-1/2} A D^{-1/2} \tag{2.6}$$

$$L_{nor2} = D^{-1} L = I - D^{-1} A \tag{2.7}$$

2.2.3 图划分准则

将给定的样本点对应为图的顶点，对两两顶点间的边赋权重值，这样便得到无向加权图 G=(V，E)，其中 V 为顶点集，E 为边的集合，此时便将聚类问题转化为对图 G 的最优划分问题。基于图论的最优划分准则：使得位于同一子图内部的样本点的相似度尽量最大，同时位于不同子图的样本点之间的相似度则尽量最小。划分准则的选取将直接影响最终的聚类结果[51]。图论中常见的划分准则有：M-cut，Average-cut，N-cut，Mbmax-cut，Ratio-cut，MN-cut 等。

1) 最小割集准则[52]

假设将图 G 划分成为 A、B 两个子图（$A \cup B = V, A \cap B = \phi$），代价函数的表达式为 $cut(A,B) = \sum_{u \in A, v \in B} w(u,v)$，其中 $w(u,v)$ 为点 u 和点 v 之间的连接权值。那么可以通过求 *mincut*（A,B）的值来对图 G 进行最优划分，即为最小割集准则。最小割集准则在图像的分割问题中的应用已经取得了较好的效果，但该准则在使用的过程中会出现歪斜分割的现象。

2) 比例割集准则[53]

该准则的目标函数如公式 2.8 所示：

$$\min Rcut = \frac{cut(A,B)}{\min(|A|,|B|)} \tag{2.8}$$

其中|A|表示子图 A 中顶点的数目，|B|表示子图 B 中顶点的数目。比例割集准则加大了类间的相似性，减少了过分割现象，但该准则会降低运行速度和运行效率。

3) 规范割集准则[16]

这一准则由 Shi 和 Malik 提出，具体的目标函数如公式 2.9 所示：

$$\min Ncut(A,B) = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)} \tag{2.9}$$

其中 $assoc(A,V)=\sum_{u\in A,t\in V} w(u,v)$，

同时 Malik 和 Shi 定义了规范关联目标函数，如公式 2.10 所示：

$$Nassco(A,B)=\frac{assco(A,A)}{assco(A,V)}+\frac{assco(B,B)}{assco(B,V)} \tag{2.10}$$

4) 平均割集准则[54]

平均割的目标函数如公式 2.11 所示：

$$\min Avcut(A,B)=\frac{cut(A,B)}{|A|}+\frac{cut(A,B)}{|B|} \tag{2.11}$$

5) 最小最大割集准则[55]

该准则的目标函数定义为：

$$\min Mcut=\frac{cut(A,B)}{assoc(A,A)}+\frac{cut(A,B)}{assoc(B,B)} \tag{2.12}$$

该方法可以有效避免孤立点的出现。

6) 多路规范准割集准则[56]

多路规范准割集准则的目的在于同时将图 G 划分为多个子图，目标函数为：

$$\min MNcut=\frac{cut(A_1,V-A_1)}{assoc(A_1,V)}+\frac{cut(A_2,V-A_2)}{assoc(A_2,V)}+\ldots+\frac{cut(A_k,V-A_k)}{assoc(A_k,V)} \tag{2.13}$$

2.3 谱聚类算法简介

根据图划分准则的不同，谱聚类算法可大致分为两大类，即多路谱聚类算法和迭代谱聚类算法，下面就这两类算法做简单介绍。

2.3.1 迭代谱聚类算法

1) PF算法

学者 Peron 与 Freeman 共同提出了 PF 算法，PF 算法主要使用数据集的相似度矩阵 W 的最大特征值所对应的特征向量 x_1 来完成聚类过程。具体的聚类方法为：特征向量中元素零对应的数据点生成一个类，而剩余的点则生成另外一个类。

2) SM算法

SM 算法指出在约束 $x^TWe = x^TDe = 0$ 条件下：

$$\min Ncut(A,B) = \min \frac{x^T(D-W)x}{x^TDx} \tag{2.14}$$

将向量 x 松弛到连续域[-1,1]上，那么求 minNcut(A,B)的问题就转化为公式 2.15：

$$\arg \min_{x^TD1=0} \frac{x^T(D-W)x}{x^TDx} \tag{2.15}$$

根据 Rayleigh 商原理，公式 2.14 的优化问题就等价与求解公式（2.16）的第二小特征值：

$$(D-W)x = \lambda Dx \tag{2.16}$$

SM 算法描述如下：

a) 利用样本集完成无向加权图 G 的构建之后，根据图 G 计算矩阵 D 与矩阵 W；

b) 求解公式$(D-W)x = \lambda Dx$次小特征值，之后求解 Fiedler 向量；

c) 在 Fiedler 向量中寻找可以使 Ncut(A，B)的值最小的划分点 i，然后将 Fiedler 向量中大于等于 Ncut(A，B)的最小值的点归为一类，其余的点则归为另一类。定义规范相似矩阵如公式 2.17 所示：

$$N = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}, \text{即 } N(i,j) = \frac{W(i,j)}{\sqrt{D(i,i)}\sqrt{D(j,j)}} \tag{2.17}$$

3) SLH算法

a) 求解相似矩阵 W 的前 k 个特征向量 $x_1, x_2, ..., x_k$，设 $X = [x_1, x_2, ..., x_k] \in R^{n \times k}$；

b) 对矩阵 X 的行向量进行规范化处理，并构建新矩阵 $Q = XX^T$；

c) 根据矩阵 Q 中的元素，将其相应的原数据点进行聚类。如果 $Q(x，y)=1$，那么点代表点 x 和点 y 在同一个类中，如果 $Q(x，y)=0$，那么代表点 x 和点 y 在不同的类中。

4) KVV算法

KVV 算法和 SM 算法原理较为相似，差异在于二者所使用的目标函数不同。Mcut 算法

M-cut 算法的目标函数，如公式 2.18 所示：

$$Mcut = \frac{x^T (D-W)x}{x^T Wx} + \frac{y^T (D-W)y}{y^T Wy} \tag{2.18}$$

对应于 2-way 划分，划分指示向量 q 如公式 2.19 所示：

$$q_i = \begin{cases} a & i \in A \\ -b & i \in B \end{cases} \tag{2.19}$$

最小化式 2.18 可得：

$$\min Mcut(A,B) = \min_q \frac{J_N(A,B)}{1 - J_N(A,B)/2} \Rightarrow \min_q J_N(A,B) \tag{2.20}$$

$$J_N(A,B) \equiv J_N(q) = \frac{q^T (D-W)q}{q^T Dq} \tag{2.21}$$

在约束条件 $x^T e = 0$ 的情况下，将向量 x 松弛至连续域[-1,1]，那么依据 Rayleigh 商原理，公式 2.21 的优化问题就可以转化为求解公式 2.22 的第二小特征值：

$$(D - W)\ x = \frac{\lambda}{1+\lambda}\ D \tag{2.22}$$

因此，M-cut 算法可以简要描述如下：

a) 通过给定的样本集构建无向加权图 G，依据图 G 计算矩阵 W 和 D；

b) 求解公式 2.22 的特征值，并获取第二小特征值对应的 Fiedler 向量；

c) 在 Fiedler 向量中查找划分点，使得该点满足 M-cut 值最小；

d) 对结果进行优化。

2.3.2 多路谱聚类算法

1) NJW算法

Ng，Jordan 等人提出用拉普拉斯矩阵的前 k 个最大特征值所对应的特征向量构造新的特征空间，新空间 R^k 中的点与原数据为一一对应的关系，然后在 R^k 空间中利用 k 均值算法进行聚类。

2) MS算法

Meila 提出了 MS 算法， MS 算法是在 NJW 算法的基础上重新构造了一个新的转移矩阵 $P = D^{-1}W$，然后用矩阵 P 的前 k 个特征值所对应的特征向量来构造新矩阵 X，再将矩阵 X 中的各行与 R^k 空间中的点一一对应，之后进行聚类。

2.4 本章小结

本章主要进行了与谱聚类相关的基础知识的介绍。首先介绍了聚类分析及其相关的经典算法，并对图的两种表现形式及相似矩阵、度矩阵及拉式矩阵进行了简单的阐述与定义，最后介绍了几种常见的经典谱聚类算法。通过对上述理论的学习，了解了谱聚类算法的基本思想及其理论基础，为后续章节中谱聚类算法的应用及改进奠定了基础。

3 一种基于密度敏感的自适应谱聚类算法

虽然基于谱聚类的许多方法在聚类的实际应用中已经取得了诸多良好的成果，但谱聚类算法的研究仍然处于发展的初期，现有的许多算法都对聚类参数特别是尺度参数十分敏感，而且需要人工手动输入聚类的数目。因此本文引入密度敏感的相似性度量，该度量是通过引入新的参数对高斯函数进行变形获得。在该度量下，属于不同类的数据点之间的距离将得到有效的放大，而属于同一类中的数据点之间的距离则缩小，最终对数据的分布情况进行有效聚类。同时本章还引入特征间隙这一概念，提出一种聚类数目自动确定的方法。数值实验的结果显示，本章所提算法可行且有效。

3.1 密度敏感的相似性度量概述

本节的分析是以传统的谱聚类算法NJW算法为基础的，NJW算法的主要步骤为：

1) 计算相似矩阵 $A\in R^{n\times n}$，其中 $A_{ii}=0$，当 $i\neq j$ 时：

$$A_{ij}=\exp\left(-\frac{d^2\left(s_i,s_j\right)}{2\sigma^2}\right) \tag{3.1}$$

2) 通过式 $L=D^{-1/2}AD^{-1/2}$ 计算拉氏矩阵，式中 $D_{ii}=\sum_{j=1}^{n}A_{ij}$，D为对角度矩阵；

3) 计算矩阵L的前k个最大的特征值，记为 $\lambda_1,...\lambda_k$，然后计算 $\lambda_1,...\lambda_k$ 所对应的特征向量 $x_1,...x_k$，（在某些情况下需要对 $x_1,...x_k$ 作正交化处理），构造新矩阵 $X=\left[x_1,x_2,...,x_k\right]\in R^{n\times k}$；

4) 规范化矩阵X的行向量，得到新矩阵Y，其中 $Y=\frac{X_{ij}}{\sqrt{\sum_j X_{ij}^2}}$；

5) 把矩阵Y的每一行与原数据集中的点进行一一对应，对这些点用FCM、c均值或其他的经典聚类算法进行聚类，可以得到k个聚类结果，具体聚类规则为：将数据点 y_i 划分到聚类j中，当且仅当Y的第i行被划分到聚类j中。

在上述算法中，有两个参数直接影响到聚类结果，相似度w和聚类数目k，两点间的

相似度w通过$\exp\left(-\frac{d^2\left(s_i,s_j\right)}{2\sigma^2}\right)$计算，其中，$d^2\left(s_i,s_j\right)$为两点间的欧氏距离，$\sigma$为尺度参数。在实际应用中，尺度参数$\sigma$的最佳取值范围有限，因此需要重复大量实验来获得最佳值，而且对于不同的数据集，σ的选取也不同。

一般情况下，聚类是一种无监督的机器学习的过程，利用数据集的一些先验知识可以提高聚类的有效性，其中最重要的就是数据集的一致性假设，即局部一致性和全局一致性。

1) 局部一致性：在空间位置上相邻的数据点有更高的相似性；

2) 全局一致性：位于同一结构上的数据点具有更高的相似性。

例如在图3.1中，有两类点，点a属于其中的一个类，点b、c、d、e属于另外的一个类。局部一致性体现为点*d*与点*b*、*e*的相似性比点*d*与点*f*、*c*的相似性高。全局一致性体现为点*c*与点*d*的相似性比点*c*与点*a*的相似性高。然而在本例中，传统的欧几里得距离只能反映数据的局部一致性，不能反映数据的全局一致性。假设在图3.1中，点*c*，*f*属于同一类，点*a*属于另一个类，则我们期望*c*，*f*之间的相似度大于*c*，*a*之间的相似度，但是在欧氏距离测度下点*c*更接近点*a*，具体参数见表3.1。

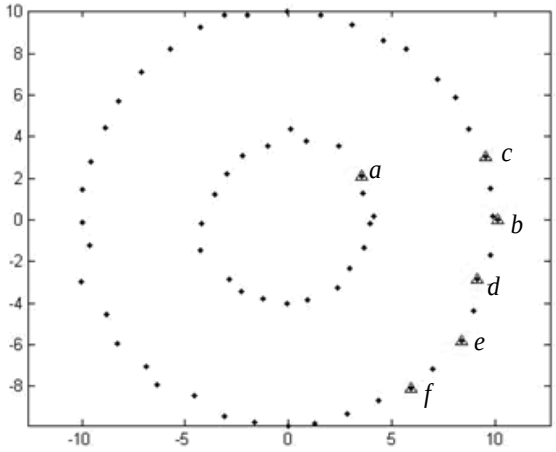


图 3. 1 不同流行上的数据点

基于以上出现的问题，我们设计一种可以同时满足局部一致性和全局一致性的相似性度量—密度敏感的相似性度量。该度量可以缩短同一类中数据点之间的距离，同时放大不同类中数据点之间的距离，有效的描述数据点的实际分布，从而达到好的聚类效果。

3.1.1 基于密度敏感的距离

定义以下密度可调节的相似性度量

$$L(x,y)=\left(e^{\rho\,dist(x,y)}-1\right)^{1/\rho} \tag{3.2}$$

在式（3.2）中，dist(*x*, *y*)表示数据点 *x* 和 *y* 之间的欧几里得距离，ρ 为密度参数，ρ 一般取大于 1 的自然数，当数据集复杂且非凸时，ρ 的值可以取的小一些。

把所给的数据点对应为一个无向加权图$G=(V,E)$的顶点集V，$V=\{x_1,x_2,...,x_n\}$，$E=\{S_{ij}\}$为两两数据点之间的相似度，由于密度敏感的相似性度量不满足三角不等式，所以不能直接用于构造相似性矩阵，因此我们根据该度量重新定义一种距离测度。

用$P=\{p_1,p_2,...,p_l\}$表示图上一顶点数为 l 的连接点 p_1 和 p_l 之间的路径，其中$p_k\in V(i=1,...,l)$，边$(p_k,p_{k+1})\in E,1\le k<l$。用$P_{ij}(1\le i,j\le n)$表示数据点对$x_i,x_j$之间的所有路径集合，定义数据点对$x_i,x_j$之间的密度敏感的距离，公式如 3.3 所示：

$$A_{ij}=\min_{P\in P_{ij}}\sum L(p_k\ p_{k+1})\quad ,k=\quad 1,\ 2,\ldots \tag{3.3}$$

其中，$L(p_k,p_{k+1})$表示两点间密度可调节的线段长度。

定理 1 对数据点$x_i,x_j,x_k\in V\ \ 1\le i,j,k\le n$，密度敏感的距离测度满足以下性质：

1) 非负性：$A_{ij}\ge 0$；

2) 自反性：$A_{ij}=0$当且仅当$x_i=x_j$；

3) 对称性：$A_{ij}=A_{ji}$；

4) 三角不等式：$A_{ij}\le A_{ik}+A_{kj}$。

证明：非负性，自反性，对称性显然成立，下面仅证明三角不等式。根据定义，点 x_i 到点 x_j 的最短距离为 A_{ij}，而 $A_{ik}+A_k$ 为点 x_i 经过点 x_k 到点 x_j 的距离，因此 $A_{ij}\le A_{ik}+A$。

重新计算图 3.1 中点 *c*, *a* 以及点 *c*, *f* 之间的距离，与欧式距离测度下的计算结果进行比较，结果如表 3.1。从表中结果可以看出，密度敏感距离更好地反应了数据的真实分布情况。

表 3.1 不同测度下的距离

线段	欧氏距离	密度敏感距离($\rho=2$)	密度敏感距离($\rho=5$)
cb	3.0626182	21.3601	21.3835
bd	3.0252648	20.5752	20.5995
de	3.0759152	21.6466	21.6697
ef	3.3230202	27.7260	27.7440
ac	6.0881400	440.600	440.701

3.1.2 相关矩阵

称 $P=D^{-1/2}SD^{-1/2}$ 为 Laplace 矩阵，其中：

$$S_{ij}=\frac{1}{A_{ij}+1} \tag{3.4}$$

$$D=\begin{bmatrix} d_1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & d_n \end{bmatrix},\ d_i=\sum_j S_{ij} \tag{3.5}$$

3.2 自适应谱聚类算法研究

传统谱聚类方法在聚类之前需要人工手动输入聚类个数，然而，在实际应用中，聚类数目k的值在正常情况下无法预先确定；同时，传统谱聚类算法中相似度的计算受参数值的影响也比较大。针对存在的这两个问题，本节通过引入两个新的概念提出了一种基于密度敏感的自适应谱聚类算法，该算法运用聚类中本征间隙之间的大小特征来实现类个数的自动确定，然后用正交特征向量对数据进行分类。

3.2.1 利用特征间隙确定聚类数目

假定在理想状态下，如果给定的数据集S存在k个可以分离的类，根据谱聚类的扰动分析理论可以证明，对于规范化相似矩阵，则有结论：矩阵的前k个最大特征值为1，同时第k+1个特征值则小于1，k个聚类的实际分布情况决定了这两个特征值之间的差值的大小。分布情况越明显，特征值的差值就越大；反之，差值越小[57]。同时，将这个差值

定义为本征间隙(Eigengap)，根据矩阵的摄动理论[57]，本征间隙的值越大，所选取的k个特征向量所构成的子空间的性质就越稳定。

记 $S'_{ij}=S_{ij}\Big/\left(\sum_j S_{ij}^2\right)^{1/2}$ 为 S 的规范化相似矩阵，设 S' 的特征值为 $\lambda_1\geq\lambda_2\geq...\geq\lambda_n$，则称 $\left\{g_1,g_2,...,g_{n-1}\middle|g_i=\lambda_i-\lambda_{i+1}\right\}$ 为特征间隙序列。

3.2.2 算法内容

输入：数据点 $V=\left\{x_1,x_2,...,x_n\right\}$；

输出：数据点的划分 $C_1,C_2,...,C_k$。

1） 根据式（3.4）构造相似矩阵 $S\in R^{n\times n}$；

2） 规范化相似矩阵 S，记为 S'，$S'_{ij}=S_{ij}\Big/\left(\sum_j S_{ij}^2\right)^{1/2}$；

3） 根据式（3.5）构造对角矩阵 D；

4） 构造拉普拉斯矩阵 $P=D^{-1/2}SD^{-1/2}$；

5） 计算规范化相似矩阵 S' 的特征值，并按大小顺序排列，记为 $\lambda_1\geq\lambda_2\geq...\geq\lambda_n$，计算特征间隙序列 $\left\{g_1,g_2,...,g_{n-1}\middle|g_i=\lambda_i-\lambda_{i+1}\right\}$，求特征间隙的最大值，记为 g_k 那么类的个数即为 k；

6） 求拉氏矩阵 P 的 k 个最大特征值所对应的特征向量 $v_1,v_2,...,v_k$，构造新矩阵 $V=\left[v_1,v_2,...,v_k\right]\in R^{n\times k}$，其中 $v_l\left(l=1,...,k\right)$ 为列向量；

7） 规范化 V 的行向量，记为矩阵 Y，其中 $Y_{ij}=V_{ij}\Big/\left(\sum_j V_{ij}^2\right)^{1/2}$；

8） 把矩阵 Y 的每一行元素看作为空间 R^k 中的一个点，通过 k 均值算法将这些点进行分类；

9） 如果矩阵 Y 的第 i 行元素属于第 j 类，那么相应的原数据点 x_i 属于第 j 类。

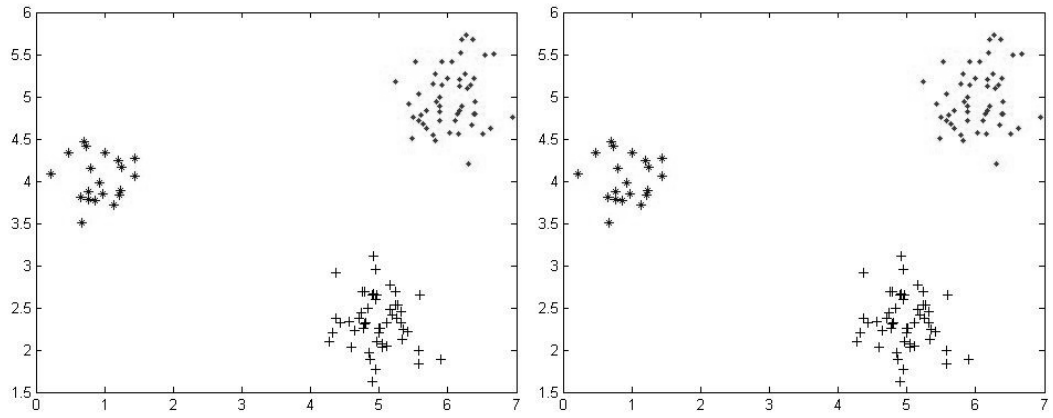
3.3 实验验证分析

为了说明本文经过改进的聚类算法在理论上分析的正确性及其在实际应用中的分类性能，我们分别针对人工数据集以及高维 UCI 数据进行了大量的实验。实验结果表明，文中提出的算法对上述各类数据集都有相对较好的聚类效果，而传统的谱聚类算法（SC）在这些数据集的聚类效果则相对较差。

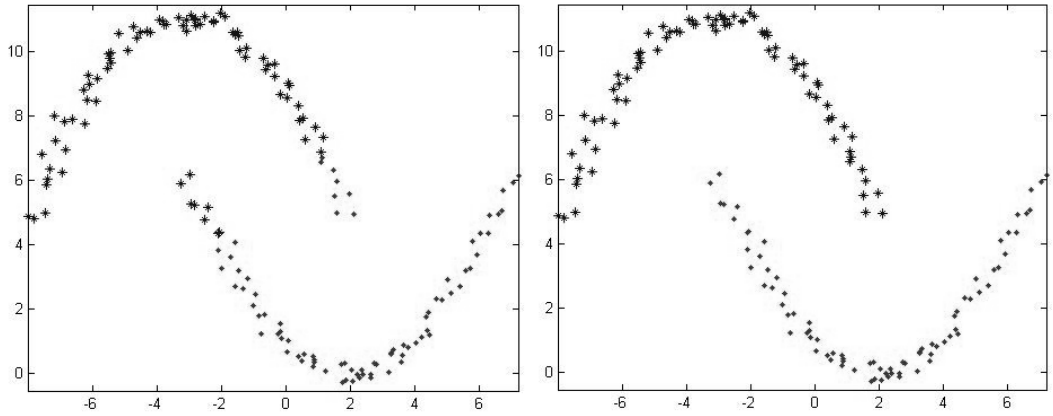
3.3.1 人工数据集

下面给出本文改进后的新算法在不同形状的数据集上的聚类结果，并且同传统的谱聚类方法(SC)的聚类结果做了比较，实验结果如图 3.2 所示，其中图 a（1）、b（1）、c（1）、d（1）为 SC 算法的聚类结果，图 a（2）、b（2）、c（2）、d（2）为本文算法结果。图 a 包含三个不规则形状的数据集，每个数据集中分别含有 20、40、40 个数据点，从分类结果可以看出，对于简单的数据集，SC 算法与本文算法都得到了理想的聚类效果；图 b 为两个交叉的抛物线形数据集，每个数据集中分别含有 80 个数据点，传统的谱聚类算法对抛物线的交叉部分得到了错误的分类结果，而本文的算法则得到了正确的分类结果；图 c 为三个圆形数据集，其中外圆为圆环，SC 算法错误地将圆环边缘与内置小圆分为一个类，图 c(2) 显示了本文的正确分类结果；图 d 为三个同心圆，SC 算法同样分类错误，本文算法分类正确。

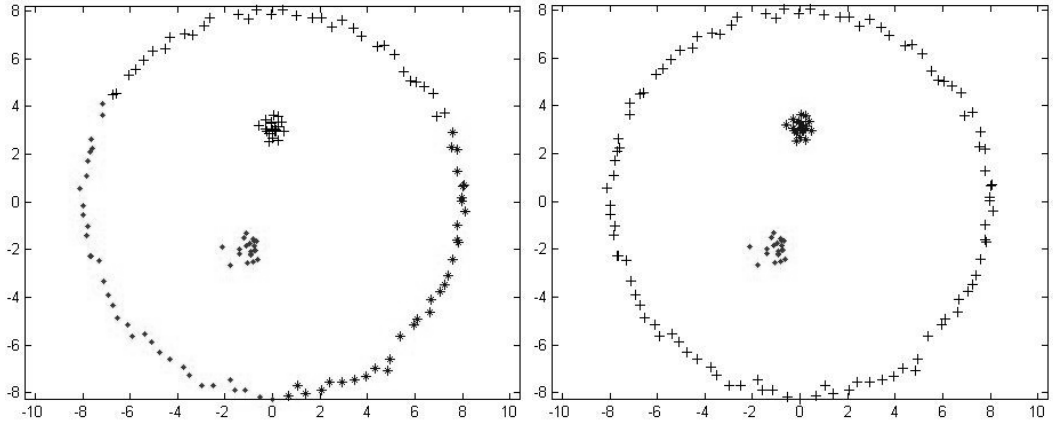
由上述实验分析可知，对于一些分类很明显的简单的数据集，SC 算法与本文算法都能得到正确的分类结果；但是对于一些复杂的数据集，例如抛物线，同心圆等，SC 算法的聚类结果出现了较大的误差。由于本文算法引入密度可调节的相似性度量，降低了不同类别数据点之间的相似度，从而得到了理想的聚类结果，并且准确地计算出了聚类数目。而 SC 算法必须手动输入聚类数目，且使用欧式距离作为相似性度量，没有准确有效的反应出数据的实际聚类分布，因此聚类效果相对较差。



a（1）　　b（1）

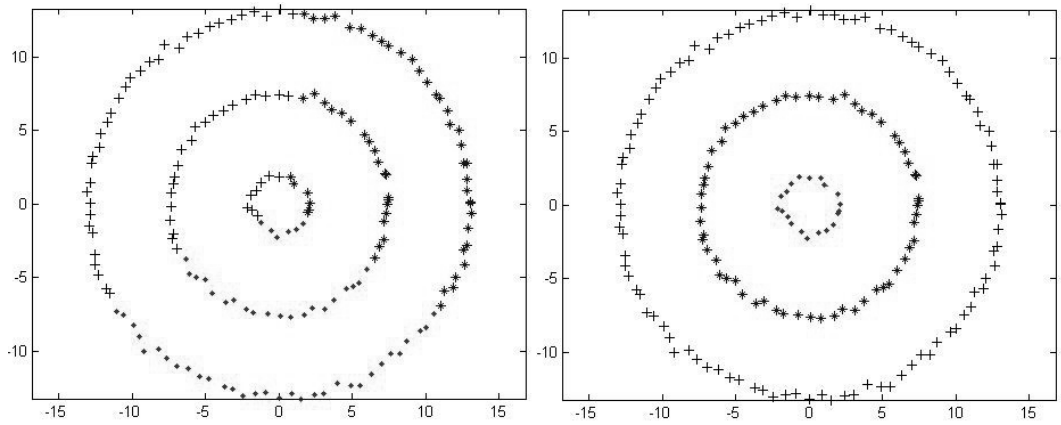


a（2）　　b（2）



a（3）　　b（3）

图 3.2　SC 算法与本文算法聚类结果比较



a（4）　　b（4）

图 3.2（续）　SC 算法与本文算法聚类结果比较

3.3.2　真实数据

本节在真实数据集上对两个算法进行的测试，并且通过对结果的对比验证本文所提谱聚类算法的聚类性能更优，那么选取 UCI 数据集上的 Iris、Glass、Teaching 作为测试数据集，其中 Iris 数据集由 150 个四维的数据点组成，分别属于三个不同的类；Glass 数据集由 214 个九维的数据点组成，分别属于六个不同的类；Teaching 数据集由 151 个五维的数据点组成，分别属于三个不同的类。

利用 SC 算法与本文算法分别对所给的三个数据集进行聚类。表 3.2 给出了这两种算法的正确率，正确率越大，聚类性能越优。

表 3.2 两种算法对 Iris、Glass、Teaching 数据集的聚类结果

数据集	SC 算法	本文算法
Iris	0.900	0.946
Glass	0.509	0.523
Teaching	0.815	0.874

从表 3.2 正确率可以看出，密度敏感的谱聚类算法明显优于 SC 算法，它可以更有效的计算出来数据的实际聚类分布情况，得到较好的聚类效果。

表 3.2 给出了本文算法与 SC 算法在 UCI 数据集 Iris、Glass、Teaching 上的运行时间，本文算法的运行时间稍大于 SC 算法，这是因为在确定聚类数目时需要一定的耗时。

表 3.3 两种算法在数据集 Iris、Glass、Teaching 上的运行时间(秒)

数据集	SC 算法	本文算法
Iris	0.397	0.411
Glass	0.892	0.964
Teaching	0.409	0.427

3.4 本章小结

本章提出一种密度敏感的自适应谱聚类算法，首先通过引入特征间隙这一概念给出了一种自动确定聚类个数的方法。其次在高斯函数的基础上定义了一种新的距离测度—密度敏感的距离测度，该测度不仅放大了不同类数据点之间的距离，同时缩小了同一类数据点之间的距离，通过新提出的算法可以实现数据的实际聚类分布的真实描述，达到了较好的聚类效果。通过人工数据集实验与真实数据实验结果可以看到，本文所提的算法比经典的谱聚类算法具有更好的聚类效果。

4 一种基于谱聚类的自适应 IPCM 算法

当给定的数据集的稀疏程度不同时， IPCM算法无法实现较理想的聚类效果，而且聚类的数目需要手工输入，针对这一问题，本章引入密度敏感距离与特征间隙两个概念，提出一种基于谱聚类的自适应IPCM算法，该算法引入一种新的聚类测度—密度敏感的距离，同时利用特征间隙的性质准确地计算出聚类的数目。数值实验验证了这一算法不仅是可行的，而且还可以有效地克服谱聚类算法和IPCM算法单独聚类时的二者的缺点。

4.1 IPCM 算法概述

模糊 c-均值聚类(FCM)[58]和可能性 c-均值聚类(PCM)[59]是聚类算法中两个经典的模型。

FCM 算法具有收敛速度较快、简单易懂、几何意义比较直观等优点，但是 FCM 算法属于局部搜索算法的范畴，因此它在最优化的过程中极容易陷入局部极小值，并且很容易受噪声点的影响。

FCM 的目标函数，如公式 4.1 所示：

$$J_{FCM}(U,V)=\sum_{i=1}^{c}\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}\,d_{ik}^{2}\left(x_k,v_i\right) \tag{4.1}$$

式中， $\boldsymbol{X}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,...,\boldsymbol{x}_N\}$ 是一组 n 维空间上的向量； $\boldsymbol{V}=\{\boldsymbol{v}_1,\boldsymbol{v}_2,...\boldsymbol{v}_c\}$ 是 c 个类的中心点集； c 为聚类个数； N 表示数据点总数； d_{ik} 表示数据点到中心点的欧氏距离； m_f 是模糊性隶属度 $u_{ik}^{(f)}$ 的权重指数，且满足 $m_f>1$。

式 4.1 约束条件，如式 4.2 所示：

$$0\le u_{ik}^{(f)}\le 1,\quad \sum_{i=1}^{c}u_{ik}^{(f)}=1(1\le k\le N),\quad 0<\sum_{k=1}^{N}u_{ik}^{(f)}\le N(1\le i\le c) \tag{4.2}$$

为了克服 FCM 算法的缺点，在 1993 年 Krishnapuram 和 J.Keller 提出了 PCM 算法。相对于 FCM 算法，其从以下两个方面进行了改进：

1） 加入约束条件；

2） 加入惩罚项η_i。

PCM算法的目标函数，如公式4.3所示：

$$J_{PCM}(U,V)=\sum_{i=1}^{c}\sum_{k=1}^{N}u_{ik}^{(p)\ m_p}d_{ik}^2+\sum_{i=1}^{c}\eta_i\sum_{k=1}^{N}(1-u_{ik}^{(p)\ m_p} \tag{4.3}$$

式中，η_i是惩罚因子，一般取为$\eta_i=\dfrac{\sum_{j=1}^{n}\left(u_{ij}^{(p)}\right)^{m_p}(d_{ij})^2}{\sum_{j=1}^{n}\left(u_{ij}^{(p)}\right)^{m_p}}$，$m_p$是可能性隶属度$u_{ik}^{(p)}$的权重指数，且满足$m_p>1$。

式 4.3 约束条件，如公式 4.4 所示：

$$0\le u_{ik}^{(p)}\le 1(1\le i\le c,1\le k\le N)；\ 0<\sum_{k=1}^{N}u_{ik}^{(p)}\le N(1\le i\le c)；\ \max_{1\le i\le c}u_{ik}^{(p)}>0(1\le k\le N) \tag{4.4}$$

PCM算法是建立在以FCM模糊初始划分的基础之上的，所以它计算出的隶属度矩阵就更加准确，PCM算法的另一个优点是可以通过减小噪声数据的隶属度值降低其对聚类中心的影响。但是，初始值的设定和惩罚因子的选取对PCM算法的聚类过程影响比较大，而且会错误地将噪声点设为聚类中心。

综合这两个典型的模型，张讲社等人提出了 FCM 与 PCM 的混合模型——IPCM 算法，该算法的目标函数如公式 4.5 所示：

$$J_{IPCM}(U^{(p)},U^{(f)},V)=\sum_{i=1}^{c}\sum_{k=1}^{N}(u_{ik}^{f})^{m_f}[(u_{ik}^{p})^{m_p}d_{ik}+\eta_i(1-u_{ik}^{p})^{m_p}] \tag{4.5}$$

式中，η_i是惩罚因子，一般取值如公式 4.6 所示：

$$\eta_i=\frac{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}d_{ik}^2}{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}} \tag{4.6}$$

在给定的约束条件下结合公式 4.2 和公式 4.4 得到所求优化问题为：$\min_{(U^{(p)},U^{(f)},V)}J_{IPCM}(U^{(p)},U^{(f)},V)$，运用 Lagrange 乘数法求解，可得到最优解的必要条件，如公式 4.7 所示：

$$u_{ik}^{(p)}=\frac{1}{1+(\frac{d_{ik}^2}{\eta_i})^{\frac{1}{m_p-1}}} \tag{4.7}$$

$$u_{ik}^{(f)}=\frac{1}{\sum_{j=1}^{c}[\frac{(u_{ik}^{(p)})^{(m_p-1)/2}d_{ik}}{(u_{jk}^{(p)})^{(m_p-1)/2}d_{jk}}]^{\frac{2}{m_f-1}}} \tag{4.8}$$

$$v_i=\frac{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}x_k}{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}} \tag{4.9}$$

IPCM 算法在处理一般聚类问题以及含有噪声点的聚类问题时都有不错的效果，但是在处理疏密程度不同或高维数据集的聚类问题时，分类效果却并不理想。

4.2 基于谱聚类的 IPCM 算法

4.2.1 算法原理

这一节我们仍使用密度敏感的距离，如公式4.10所示：

$$A_{ij}=\min\sum_{P\in P_{ij}}L(p_k,p_{k+1}),\quad k=1,2,...,l-1 \tag{4.10}$$

构造相似矩阵与拉普拉斯矩阵，其中$L(x,y)=\left(e^{\rho\mathrm{dist}(x,y)}-1\right)^{1/\rho}$，dist($x$, y)为数据点 x 和 y 之间的欧几里得距离，$\rho>1$为密度参数。$P=\{p_1,p_2,...,p_l\}$表示图上一顶点数为l的连接点 p_1 和 p_l 的路径，其中边$(p_k,p_{k+1})\in E,1\le k<l$，则 $P_{ij}\left(1\le i,j\le n\right)$表示连接数据点对 x_i,x_j 的所有路径集合。

定义：记 $W'_{ij}=W_{ij}\Big/\left(\sum_j W_{ij}^2\right)^{1/2}$ 为 W 的规范化相似矩阵，设 W' 的特征值为 $\lambda_1\ge\lambda_2\ge...\ge\lambda_n$，则称$\{g_1,g_2,...,g_{n-1}|g_i=\lambda_i-\lambda_{i+1}\}$为特征间隙序列。

用式 4.10 代替式 4.5 中的欧氏距离，即得到新的目标函数，如公式 4.11 所示。

$$J(U^{(p)},U^{(f)},V)=\sum_{i=1}^{c}\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}[(u_{ik}^{(p)})^{m_p}A_{ik}+\eta_i(1-u_{ik}^{(p)})^{m_p}] \tag{4.11}$$

其中，

$$\eta_i = \frac{\sum_{k=1}^{N} (u_{ik}^{(p)})^{m_p} (u_{ik}^{(f)})^{m_f} A_{ik}}{\sum_{k=1}^{N} (u_{ik}^{(p)})^{m_p} (u_{ik}^{(f)})^{m_f}} \tag{4.12}$$

公式 4.11 约束条件，如公式 4.13 所示：

$$0 \le u_{ik}^{(f)} \le 1, \quad \sum_{i=1}^{c} u_{ik}^{(f)} = 1 (1 \le k \le N), \quad 0 < \sum_{k=1}^{N} u_{ik}^{(f)} < N (1 \le i \le c),$$

$$0 \le u_{ik}^{(p)} \le 1 (1 \le i \le c, 1 \le k \le N), \quad 0 < \sum_{k=1}^{N} u_{ik}^{(p)} < N (1 \le i \le c), \quad \max_{1 \le i \le c} u_{ik}^{(p)} > 0 (1 \le k \le N) \tag{4.13}$$

定理：最小化 J 的必要条件为

$$u_{ik}^{(p)} = \frac{1}{1 + (\frac{A_{ik}}{\eta_i})^{\frac{1}{m_p - 1}}} \tag{4.14}$$

$$u_{ik}^{(f)} = \frac{1}{\sum_{j=1}^{c} [\frac{(u_{ik}^{(p)})^{(m_p - 1)/2} A_{ik}}{(u_{jk}^{(p)})^{(m_p - 1)/2} A_{jk}}]^{\frac{2}{m_f - 1}}} \tag{4.15}$$

$$v_i = \frac{\sum_{k=1}^{N} (u_{ik}^{(p)})^{m_p} (u_{ik}^{(f)})^{m_f} x_k}{\sum_{k=1}^{N} (u_{ik}^{(p)})^{m_p} (u_{ik}^{(f)})^{m_f}} \tag{4.16}$$

公式 4.14 中，惩罚因子 η_i 取值同公式 4.12。

证明：运用 Lagrange 乘数法可知

$$J(U^{(p)}, U^{(f)}, V) = \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik}^{(f)})^{m_f} [(u_{ik}^{(p)})^{m_p} A_{ik} + \eta_i (1 - u_{ik}^{(p)})^{m_p}] \tag{4.17}$$

$u_{ik}^{(f)}$、$u_{ik}^{(p)}$ 的计算只需将其中的 d_{ik}^2 换成 A_{ik} 即可。

下面以 v_i 的证明为例：

由于 $\frac{\partial J}{\partial v_i} = 2 \sum_{k=1}^{N} (u_{ik}^{(f)})^{m_f} (u_{ik}^{(p)})^{m_p} A_{ik} \cdot \frac{x_k - v_i}{\sigma^2}$

令公式 $\frac{\partial J}{\partial v_i} = 0$ 可得：

$$2\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}(u_{ik}^{(p)})^{m_p}A_{ik}\cdot\frac{x_k}{\sigma^2}=2\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}(u_{ik}^{(p)})^{m_p}A_{ik}\cdot\frac{v_i}{\sigma^2}$$

$$\Rightarrow v_i=\frac{\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}(u_{ik}^{(p)})^{m_p}A_{ik}\cdot x_k}{\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}(u_{ik}^{(p)})^{m_p}A_{ik}}$$

$$\Rightarrow v_i=\frac{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}x_k}{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}} \quad (4.18)$$

4.2.2 算法内容

输入：数据点集 $\boldsymbol{X}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,...,\boldsymbol{x}_N\}$，设定迭代的停止阈值 $\varepsilon_1>0,\varepsilon_2>0$，设置迭代计数器 $p=0$。

输出：数据点的划分 $V_1,V_2,...,V_k$。

1) 构造相似矩阵 W，其中 $W=\min\sum_{p\in P_{ij}}\left(\exp\left(\rho\mathrm{dist}\left(p_k,p_{k+1}\right)\right)-1\right)^{1/\rho}$；

2) 规范化相似矩阵 W，记为 W'，$W'_{ij}=W_{ij}\Big/\left(\sum_j W_{ij}^2\right)^{1/2}$；

3) 构造对角矩阵 $D=\begin{bmatrix}d_1 & & \\ & \ddots & \\ & & d_n\end{bmatrix}$，$d_i=\sum_j S_{ij}$，拉普拉斯矩阵 $P=D^{-1/2}SD^{-1/2}$；

4) 计算规范化相似矩阵 W' 的特征值，并按大小顺序排列，记为 $\lambda_1\geq\lambda_2\geq...\geq\lambda_n$，计算特征间隙序列 $\{g_1,g_2,...,g_{n-1}\,|\,g_i=\lambda_i-\lambda_{i+1}\}$，在计算的特征间隙中寻找第一个极大值，类那么的个数即为该极大值对应的下标，记为 k；

5) 用 FCM 的聚类结果初始化聚类原型 $V^{(0)}$ 和 $U^{(f)}$；

6) 初始化 η_i，$\eta_i=\dfrac{\sum_{j=1}^{n}\left(u_{ij}^{(p)}\right)^{m_p}(d_{ij})^2}{\sum_{j=1}^{n}\left(u_{ij}^{(p)}\right)^{m_p}}$；

7) 用式 4.14 和公式 4.15 计算或更新可能性划分矩阵$\tilde{U}^{(p)}=(\tilde{u}_{ij}^{(p)})_{k\times N}$和模糊划分矩阵$U^{(f)}=(u_{ij}^{(p+1)})_{k\times N}$；

8) 用式 4.6 更新聚类原型$V^{(p+1)}=\left(v_1^{(p+1)},v_2^{(p+1)},\cdots,v_k^{(p+1)}\right)$；

9) 用一个矩阵范数$\|\bullet\|$来比较$V^{(p)}$与$V^{(p+1)}$，若$\left\|V^{(p)}-V^{(p+1)}\right\|>\varepsilon_1$，让$p=p+1$，转向步骤（7）；

10) 若$\left\|V^{(p)}-V^{(p+1)}\right\|\geq\varepsilon_2$，用式 4.12 重新估计$\eta_i$，再执行步骤（7）~步骤（9），直到$\left\|V^{(p)}-V^{(p+1)}\right\|<\varepsilon_2$，输出最后得到的划分矩阵$U$、$\tilde{U}$和聚类原型$V$。

4.3 数值试验

为了说明本文所提聚类算法相较于 IPCM 算法有更好的分类性能，我们分别针对人工数据集以及高维数据进行了大量的实验，并同 IPCM 算法做了比较。

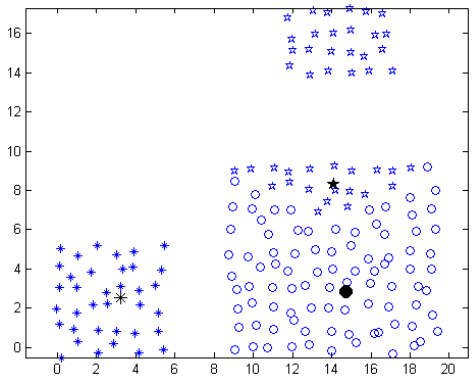
4.3.1 人工数据集

下面通过三个模拟实验来验证本文算法可以准确选择聚类中心、计算聚类数目，并同 IPCM 算法做了比较。实验结果如图 4.1 所示，其中，图 a（1）、a（2）、a（3）为 IPCM 算法的聚类结果，图 b（1）、b（2）、b（3）为本文算法的聚类结果。参数的设定如下：$\varepsilon_1=\varepsilon_2=10^{-6}$，$m_f=2$，$m_p=2$，$\sigma^2=2.0$。

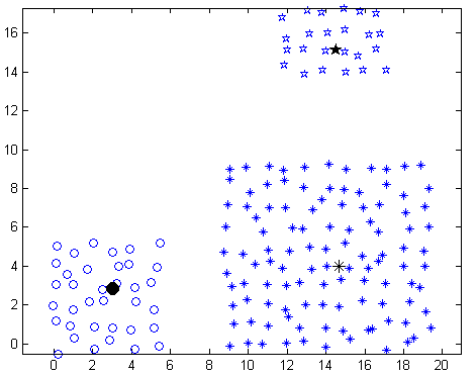
模拟数据采用二维数据，第一个数据集 S1 包含三个类，分别来自三个二维正态总体的随机采样，协方差矩阵都为$\begin{bmatrix}0.4 & 0\\ 0 & 0.4\end{bmatrix}$，每个类含有的数据点数分别为 24，36，100；第二个数据集 S2 包含四个类，分别来自四个二维正态总体的随机采样，协方差矩阵都为$\begin{bmatrix}0.2 & 0\\ 0 & 0.2\end{bmatrix}$，每个类含有的数据点数分别为 30，20，40，10；第三个数据集 S3 包含四个类，分别来自于四个二维正态总体的随机采样，协方差矩阵都为$\begin{bmatrix}0.1 & 0\\ 0 & 0.1\end{bmatrix}$，每个类含有的数据点数分别为 20，20，20，50。所有数据集的真实的类中心见表 4.1。

从图 4.1 中可以看出，IPCM 算法在计算聚类中心时出现偏差，从而导致了分类结果错误，尤其在比较复杂的数据集 S3 中，外部大圆周与内部实心类错误的分到了一类。而本文算法较为准确的计算出了聚类的中心，从而得到正确的分类结果。表 4.1 给出了 IPCM 算法与本文算法计算的中心点的结果，并且将计算结果与实际的中心点进行了对比，从表中的对比数据可以看出，本文算法的计算结果更接近于实际值。

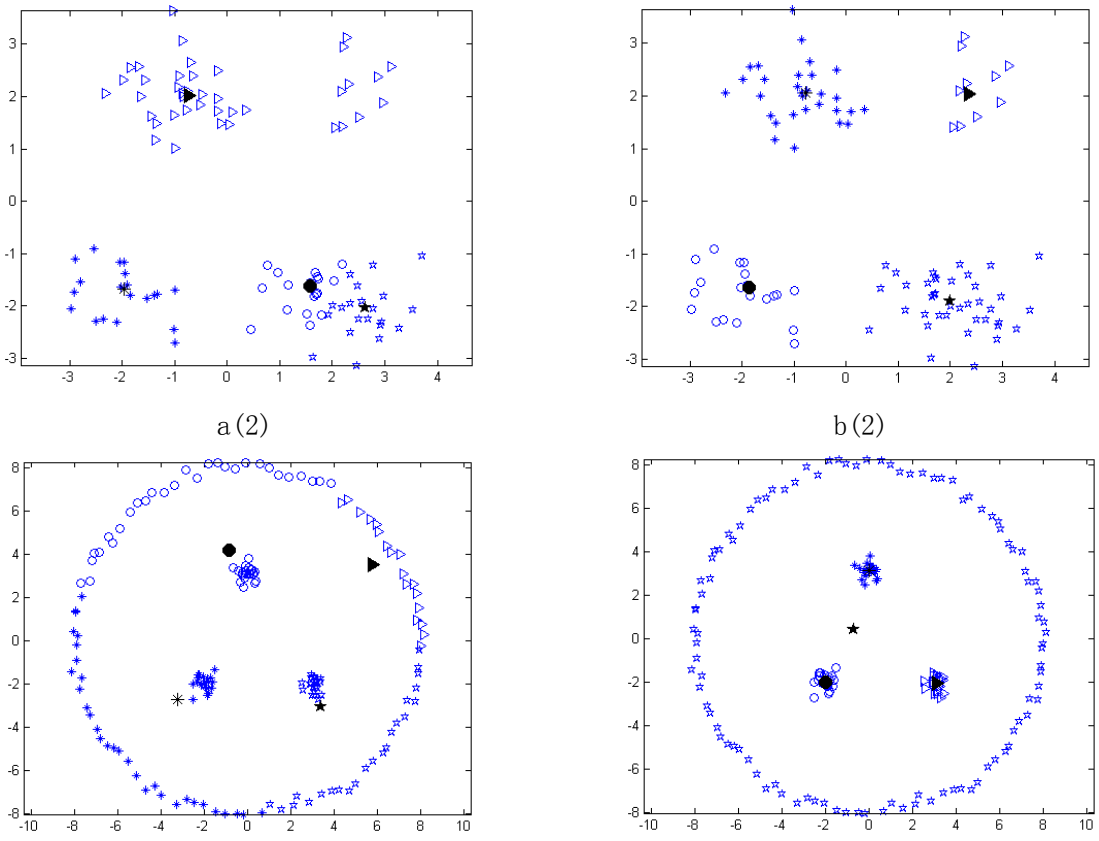
为了证明本算法的在分类过程中的稳定性和精确性，并且具有可以自动确定聚类数目这一优点，分别在给定的三个数据集上各运行 100 次，每一次都用本文算法计算聚类数目，得到了准确计算出聚类数目的概率分别为 95%，95%，90%。



a(1)



b(1)



a(2)　　b(2)

a(3)　　b(3)

图 4.1　PICM 算法与本文算法聚类结果比较

表 4.1　IPCM 算法与本文算法聚类结果比较

数据集	中心点			分类正确率	
	原中心点	IPCM 算法	本文算法	IPCM 算法	本文算法
S1 (N=160)	(3, 3) (15,4.5) (15,16)	(3.312,2.875) (14.867,2.903) (14.173,8.206)	(3.027, 2.983) (14.891, 4.312) (14.892,15.649)	87.5%	100%
S2 (N=100)	(-2,-2) (-1,-2)	(-1.973,-2.836) (-0.873,-2.025)	(-1.934,-2.364) (-0.927, 2.001)	84.0%	100%

	(2, -2) (2 ,2)	(2.701, -2.012) (1.516, -1.863)	(2.008, -1.963) (2.201, 2.034)		
S3 (N=110)	(-2,-2) (3,-2) (0, 0) (0, 3)	(-3.219,-3.107) (3.741, -3.206) (5.714, 3.210) (-1.134, 4.057)	(-2.011,-2.034) (3.102, -2.021) (-0,124,0.147) (0.093, 3.012)	54.5%	100%

4.3.2 真实数据

为了验证本文所提聚类算法的聚类性能，全区 UCI 数据集上的 Iris、Teaching、wine 作为测试的数据集，其中 Iris 数据集由 150 个四维的数据点组成，分别属于三个不同的类；Teaching 数据集由 151 个五维的数据点组成，分别属于三个不同的类；wine 数据集由 178 个十三维的数据点组成，分别属于三个不同的类。各参数的取值为：$\varepsilon_1 = \varepsilon_2 = 10^{-6}$，$m_f = 2$，$m_p = 2$，$\sigma^2 = 2.3$。

分别应用 IPCM 算法与本文算法对以上三个给定的数据集进行聚类。表 4.2 中给出了这两种算法的分类正确率，正确率越大，则聚类性能就越优。可以看出，与 IPCM 算法相比，本文算法不仅可以自动确定聚类的数目，而且分类的正确率也更高。

表 4.2 两种算法对 Iris、Glass、Teaching、wine 数据集的聚类正确率

数据集	IPCM	本文算法
Iris	0.9333	0.9533
Teaching	0.8940	0.9072
wine	0.7360	0.8314

4.4 结论

将本文所给的基于密度敏感的谱聚类算法同经典的IPCM算法相结合，提出一种基于谱聚类的自适应IPCM算法，新算法有效的克服了IPCM算法单独聚类时的缺点。通过在人工数据集实验与真实数据实验的实验结果可以看到，本文改进后的的算法比经典的IPCM算法有更好的聚类效果。

5 基于密度敏感的 Dcut 单阈值图像分割法

阈值法是图像分割中一种应用非常广泛的经典算法，该算法由于计算简单，所以运算速度快，效率高。算法的关键步骤在于阈值的确定，本章在前几章的基础上提出了一种新的图像阈值分割法——基于谱聚类的图像阈值分割法：通过引入新的相似性度量来定义一种新的节点权值计算方法，同时采用新的相似度函数定义节点之间的相似度，构建基于灰度级的相似矩阵，然后利用判别割Dcut算法对节点进行分类，再根据分类结果来确定图像分割的阈值，对图像进行分割。实验表明，该方法分割图像用时少，与现有的经典的单阈值分割方法相比，具有更为优越的分割性能。

5.1 判别割 Dcut 算法

5.1.1 算法概述

给定一幅待分割的图，由谱聚类算法的思想可知，图像的像素即组成数据样本点，根据图论的相关理论，每个数据点对应为图的顶点，两两顶点构成一条边，边的权重视为样本间的相似度，于是得到无向加权图G=(V,E,S)。设$V=\left(v_1,v_2,...,v_n\right)$为顶点的集合，E为顶点间边的集合，$s_{ij}\left(s_{ij}>0\right)$为边的权值，反映了节点$v_i$和$v_j$的相似程度，很显然$S=\left(s_{ij}\right)$为对称矩阵。基于以上的设定，对图的分割问题即转化为对顶点集合V的划分问题，如果要将V分成两个独立的子集A和B，其中B=V-A，那么把连接集合A和集合B中所有节点的边删除，就可以得到集合A和集合B的分离度，记为割（cut），如公式5.1所示：

$$cut\left(A,B\right)=\sum_{i\in A,j\in B}s_{ij} \tag{5.1}$$

Wu和Leahy通过对最小划分准则的改进，提出了Min-cut算法，但是Min-cut算法易将图中的孤立点独立的划分为一个类，为了避免这一问题，Shi和Malik提出了规范割集准则N-cut，N-cut的定义如公式5.2所示：

$$Ncut(A,B)=\frac{cut(A,B)}{assoc(A,V)}+\frac{cut(B,A)}{assoc(B,V)} \tag{5.2}$$

式中，$assoc(A,B)=\sum_{i\in A,\, j\in V} s_{ij}$，在Ncut的基础上Chen和Feng人等提出了判别割Dcut准则，Dcut定义如公式5.3所示：

$$Dcut(A,B)=\frac{cut(A,A)}{cut(A,B)+\mu|A|}+\frac{cut(B,B)}{cut(B,A)+\mu|B|} \tag{5.3}$$

式中，$\mu>0$，是常数常数；$|A|$和$|B|$分别为集合A、B中的点的数目。

对应于图论知识，图G的最优划分问题就转化为求Dcut的最小值，这时，最小化式5.3等价于最小化式5.4：

$$L^{-1}S\xi=\lambda\xi \tag{5.4}$$

式5.4中，$L=D-S+\mu I$，D是对角矩阵，对角线上的元素为$d_{ii}=\sum_j s_{ij}$的，λ和ξ即为相应的特征值和特征向量。

5.1.2 算法内容

输入：点集$V=(v_1,v_2,...,v_n)$，$v_i\in R^D$，R^D为D维空间，k,σ为参数。

输出：k个聚类。

1) 定义距离$d(v_i,v_j)$，计算相似矩阵$S=(s_{ij})\in R^{n\times n}$，其$s_{ij}=\exp\left(-d^2(v_i,v_j)/2\sigma^2\right)$；
2) 计算矩阵D、L、W，其中$W=L^{-1}S$；
3) 求矩阵W的前k个最大特征值，并按大小顺序排列，记为$\lambda_1\geq\lambda_2\geq...\geq\lambda_n$，特征值对应的特征向量记为$u_1,u_2,...,u_k$，设$U_0=[u_1,u_2,...,u_k]$；
4) 单位化处理矩阵U_0的每一行，得矩阵U；
5) 将矩阵U的每一行元素对应于空间R^k中的一个点，通过k-means算法对这些点进行分类；
6) 如果U的第i行分在第j类，那么顶点v_i属于第j类。

5.2 密度敏感的 Dcut 单阈值图像分割法

5.2.1 算法原理

在 Dcut 算法中，节点之间的相似性是通过高斯核函数计算，由于本文第三章提出的密度敏感的相似性度量经验证可以更好的描述数据的实际分布情况。本章依旧引入密度敏感的相似性度量改进 Dcut 算法，并将改进后的算法应用于图像分割。

密度敏感的距离的定义，如公式5.5所示：

$$L(x,y)=\left(\mathrm{e}^{\rho \mathrm{dist}(x,y)}-1\right)^{1/\rho} \tag{5.5}$$

式 5.5 中，dist(*x*, *y*)为点 *x* 和 *y* 之间的欧几里得距离，$\rho>1$为密度参数。

设 $I=f\left[(i,j)\right]_{M\times N}$ 是尺度为 $M\times N$ 的一幅图像，其中 *f(i,j) (i=0,1,...M-1,j=0,1,...,N-1)* 为图像在像素（i,j）处的灰度值。

设 $L=\{0,1\ldots,255\}$

$$P=\left\{(i,j)\middle| i=0,1,...M-1,\ j=0,1,...N-1\right\}$$

则 $f(x,y)\in L,\forall(x,y)\in P$

设 H=[h(k)](k=0,1,…,255)为图像的灰度统计直方图频数矩阵，其中，

$$h(k)=\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\delta_{ij}(k) \tag{5.6}$$

$$\delta_{ij}(k)=\begin{cases}1 & if\ \ f(i,j)=k\\ 0 & else\end{cases}\qquad k=0,1,...,255 \tag{5.7}$$

直方图对应在二维平面上的点集记为$V=\left\{\left(k,h(k)\right)\middle| k=0,1,...,255\right\}$。

设$v_k=\left(k,h(k)\right)$表示一个点，用一条边将每对点进行连接，对每条边赋权重值。建立加权无向图 G=(V,E,S)，设v_i，v_j是图 G 的两个节点，节点权值的定义如公式 5.8 所示：

$$w\left(v_i,v_j\right)=\left(\mathrm{e}^{\rho \mathrm{dist}\left(v_i,v_j\right)}-1\right)^{1/\rho} \tag{5.8}$$

5.2.2 算法步骤

基于密度敏感的Dcut单阈值图像分割法：

1) 求解计算直方图H（采用公式5.6）；

2) 计算相似度矩阵W（采用公式5.8）；

3) 规范化相似矩阵W，记为W'，$W'_{ij}=W_{ij}\Big/\left(\sum_j W_{ij}^2\right)^{1/2}$；

4) 构造对角矩阵$D=\begin{bmatrix} d_1 & & & \\ & \ddots & & \\ & & \ddots & \\ & & & d_n \end{bmatrix}, d_i=\sum_j W_{ij}$，拉氏矩阵$P=D^{-1/2}WD^{-1/2}$；

5) 求解P的特征值及其特征向量，从中计算出P的两个最大的特征值对应的特征向量v_1，v_2，构造矩阵$V=[v_1,v_2]\in R^{n\times 2}$；

6) 规范化矩阵V的行向量，得到矩阵新矩阵Y，其中$Y_{ij}=V_{ij}\Big/\left(\sum_j V_{ij}^2\right)^{1/2}$；

7) 将矩阵Y的每一行元素对应于空间R^2内的一点，使用k-mean法对其进行分类；

8) 若Y的第i行属于第j类，那么对应的原数据点v_i便属于第j类；

9) 根据分类结果，计算Dcut的最小值，作为分割阈值，分割图像。

5.3.3 算法验证

在实验中，先对每一幅图像进行人工手动分割，在理想的情况下通过在图像中获得的目标得到图像的最佳分割阈值，之后用该阈值进行图像分割，假定该图像得到的目标像素的个数n_0，并以人工分割的结果为标准将其他阈值分割算法与其进行对比。

假设通过其他的阈值分割方法得到的目标像素的个数为n_i，n_i^{diff}表示第i种阈值分割方法的绝对误差，那么$n_i^{diff}=|n_i-n_0|$。图像总的像素个数记为N，用$r_i^{\sigma r}$表示第i种阈值分割方法得到的绝对误差率，那么有$r_i^{\sigma r}=\dfrac{|n_i-n_0|}{N}\times 100\%$。

本章将选择三种经典的图像阈值分割方法与本文算法得分割结果进行试验对比，即最小误差法（Kittler方法）、最大熵法（Kapur方法）和最大类间方差法（Otsu 方法），

实验结果如图5.1所示：

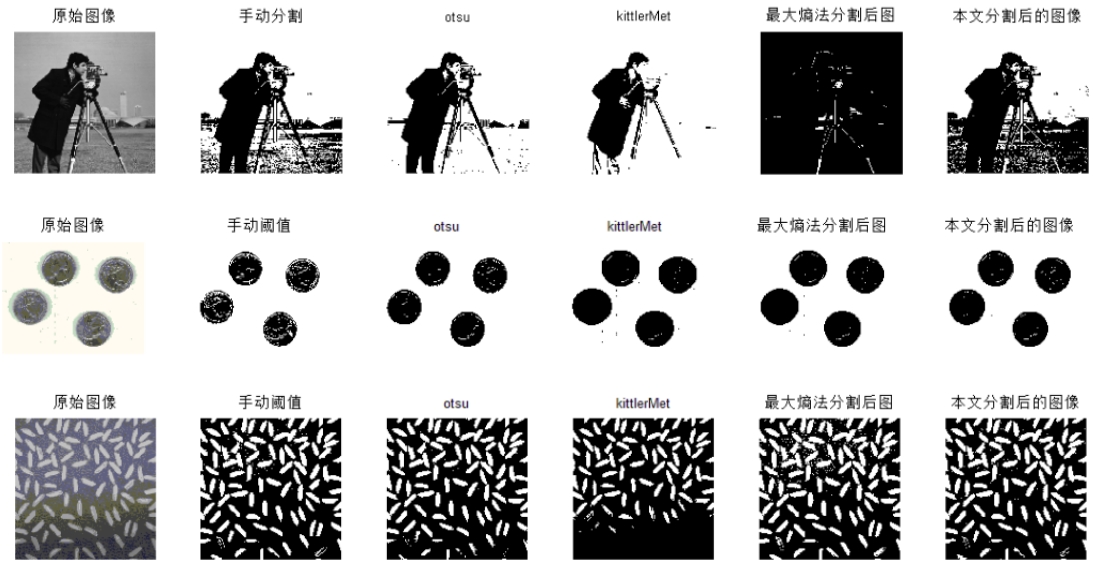


图 5.1　实验分割图

由实验分割图可以获知，对给定的三幅灰度图，本文改进的方法都可以有效地提出目标，且计算得到的分割阈值较其他三种方法更接近手动分割阈值。从表5.1看出，相对于其他算法本文方法的平均误差率最低，为8.98％．其中Kitter方法的误差率最高，达到43.40％。

表 5.1 各种方法的分割阈值和误差率

图像	图 3.2（a）		图 3.3（a）		图 3.4（a）		平均误差
	阈值	误差	阈值	误差	阈值	误差	
人工分割	145	0	75	0	110	0	0
Kitter 方法	110.4915	0.23799	52.22	0.303733	26.36	0.760364	0.434029
Kapur 方法	164	0.131034	120	0.6	178.94	0.626727	0.452587
Ostu 方法	112.9905	0.220755	90.55	0.207333	87.49	0.204636	0.210908
本文方法	128	0.117241	77.55	0.034	123	0.118182	0.089808

5.3 本章小结

将密度敏感的距离引入至判别割算法中代替原有的欧式距离，同时对判别割算法进

行改进，提出一种基于密度敏感和判别割的图像单阈值分割法。相对于其他基于图的图像分割方法，该方法在算法所需存储空间以及算法的实现复杂度均有了较大的性能提高，算法更加合理有效。

6 总结与展望

6.1 总结

本文主要对谱聚类的算法及其主要应用进行了学习和研究，主要的工作如下：

1) 对谱聚类算法进行了系统的学习，了解了关于谱聚类分析的相关基础知识，分析了谱聚类算法中将来研究中所需要突破的几个关键技术；
2) 针对传统的谱聚类算法中的两个主要的问题——如何定义相似矩阵以及如何自动确定类的数目问题，本文引入了密度敏感的距离和特征间隙两个概念，在此基础上提出了一种基于密度敏感的自适应谱聚类算法。之后在人工数据集上及UCI数据集上验证了所提算法的可行性和有效性。
3) 针对在稀疏程度不同的数据集上IPCM算法聚类效果不够理想，而且需要手动输入聚类数目的缺点。本文引入两个新的概念：密度敏感的距离与特征间隙，在此基础上对经典IPCM算法进行改进，提出新算法，该算法用密度敏感的距离代替传统的欧氏距离，并通过特征间隙准确地计算了聚类数目。数值实验验证了新算法的可行性以及有效性，同时该算法可以有效的克服IPCM算法和谱聚类算法单独聚类时的缺点。
4) 在判别割算法中用密度敏感的相似性度量代替欧几里得距离，提出一种基于密度敏感的判别割的图像阈值分割法。新算采用基于灰度级的权值矩阵描述图像像素的关系，相对于其他基于图的图像分割方法，该方法在算法所需存储空间以及算法的实现复杂度均有了较大的性能提高。

6.2 展望

本文对谱聚类算法进行了一定的学习和研究，鉴于个人能力有限，笔者认为在以后的工作中，还需要对论文从以下几方面进行改进和完善：

1) 研究适合构造相似度函数的通用规则；
2) 改进谱聚类的运行效率，缩短运行时间，有效实现谱聚类算法在海量数据中的应用；

3) 如何结合数据集所给的已知分类信息实现分类，实现基于半监督学习的谱聚类算法，也需要进一步探索研究；

4) 如何根据实际应用中的问题，提高分类的准确性，同样需要进一步的研究。

参考文献

[1] C. T. Zahn. Graphic theory for detecting and describing gesalt cluster[J], IEEE Trans Computer. 1971, 20(5):70-89.

[2] Maurizio F, Francesco C, Francesco M, Stefano R, A survey of kernel and spectral methods for clustering [J].Pattern Recognition 2008, 41 (3) :176 – 190.

[3] 施蓓蓓，郭玉堂，胡玉娟，俞骏著.多尺度的谱聚类算法[J].计算机工程与应用，2011,47(8):128-132.

[4] Fisher D H. Knowledge acquisition via incremental conceptual clustering [J]. Machine Learning, 1987, 2(2):139-172.

[5] 赵凤，焦李成，刘汉强，公茂果.半监督谱聚类特征向量选择算法[J].模式识别与人工智能，2011,24(1): 48- 55.

[6] 孙吉贵，刘杰，赵连宇. 聚类算法研究[J].软件学报, 2008,19(1):48-61.

[7] Distell, Graph theory [M]. Beijing: World Publishing Corporation, 2008.

[8] Weifu Che, Guocan Feng. Spectral clustering: A semi-supervised approach. Neuro computing [J].2012,77(1).119-228.

[9] Zhao F, Liu H, Jiao L. Spectral clustering with fuzzy similarity measure [J]. Digital Signal Processing.2011,21(6):56-63.

[10] Carlos Alzate, Johan A K, Suykens. Hierarchical kernel spectral clustering [J]. Pattern Recognition 2012,35(3):24-35.

[11] 于剑. 机器学习及其应用[M].北京：清华大学出版社，2007.

[12] Ulrike Luxburg. A tutorial on spectral clustering [J]. Statistics and Computing, 2007, 17 (4):395-416.

[13] 蔡晓妍，戴冠中，杨黎斌.谱聚类算法综述 [J]. 计算机科学,2008,35(7):14-18.

[14] Filipponea M, Camastrab F, Masullia F, Rovctta S. A survey of kernel and spectral methods for clustering[J].Pattern Recognition,2008,41(1):176-190.

[15] Luxburg U. A tutorial on spectral clustering[J]. Statistics and Computing, 2007,17(4): 395-416.

[16] Shi J, Malik J. Normalized cuts and image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(8):888-905.

[17] Scott G L, Longuet H C. Feature grouping by re-localization of eigenvectors of the proximity matrix [C] Proc British Machine Vision Conference.1990:103-108.

[18] Kannan R, Vempala S, Vetta A. On clustering-good, bad and spectral [J]. FOCS，2000：367-377.

[19] 田铮，李小斌，句彦伟. 谱聚类的扰动分析[J]，中国科学，2007,34(7):527-543.

[20] Ng A Y, Jordan M I, Weiss Y, On spectral clustering: analysis and an algorithm [J]. Advance in Neural Information Processing Systems, 2002, 2: 849-856.

[21] Meila M，Shi J. Learning segmentation by random walks[J]. In NIPS，2000:873-879.

[22] David T, Shafagh F, Joseph B, A spectral clustering method for microarray data [J]. Computational Statistics and Data Analysis 2005,49 (6): 63-76.

[23] Jain A, Murty M. Flynn P. Data clustering：A review[J].ACM Computing Surveys. 1993,31(3):264-323.

[24] Jain A K, Flynn P J. Image segmentation using clustering.[J] Advances in Image Understanding. 1996,65-83.

[25] 王玲，薄列峰，焦李成.密度敏感的半监督谱聚类[J].软件学报,2007,18(10):2412-2422

[26] Huang Z, Extensions to the k-means algorithm for clustering large data sets with categorical values [J]. ACM SIGMOD Record, 1999, 28(2): 49-60

[27] Chen W F, Feng G C. Spectral clustering with discriminate cuts [J]. Knowledge-Based Systems, 2012,28(7):27-37.

[28] 卫俊霞，相里斌，高晓惠. 段晓峰.基于 K 均值聚类与夹角余弦法的多光谱分类算法[J].光谱学与光谱分析,2011.31(5):1357-1360.

[29] Qiu H J, Edwin R. Graph matching and clustering using spectral partitions [J].Pattern Recognition, 2006,39(4):22-34.

[30] Alzate C, Suykens J. A. K. Hierarchical kernel spectral clustering[J]. Neural Networks (2012), doi:10.1016/j.neunet.2012.06.007.

[31] Qin G M, Gao L, Spectral clustering for detecting protein complexes in protein-protein interaction (PPI) networks [J].Mathematical and Computer Modeling. 2010(52) 2066-2074.

[32] Dempster A P, Laird N M, Rubin D B. Maximum likelihood from In complete data vis the EM algorithm [J]. Journal of Royal Statistical Society Series, 1997, 39(1):1-38.

[33] Marques JP 主编，吴逸飞译. 模式识别原理、方法及应用[M].北京：清华大学出版社, 2005.

[34] Ng R，Hart J. Efficient and effective clustering methods for spatial datamining[J] Proc.20thInf．Conf. Very Large Databases, Santiago, Chile, Morgan Kaufmann. 1994, 144-155.

[35] Kufman K L, Roueeeuw P J. Finding groups in data：An introduction to cluster analysis [J]. Pattern Recognition, 2005, 33(7):87-96.

[36] Wang W, Yang J, Muntz R. A statistical information grid approach to spatial data mining[J]．Proc．23rd Int．Conf．on VLDB, Morgan Kaufmann,1997,186—195.

[37] Sheikholeslami G, Chatterjee S, Zhang A. Wave cluster：A multi-resolution clustering approach for very large spatial databases[J]Proc 24th In．Conf．on very large databases, New York. 1998, 428-439.

[38] Guha S, Rastogi R, Shim K. Cure：An efficient clustering algorithm for large database [J].ACM SIGMOD, 1998, 26(1):73-84.

[39] Guhu S, Rastogi R, Shim K. Rock：A robust clustering algorithm for categorical attributes．Information Systems, 2000,25(5):345-366.

[40] Fisher D. Knowledge acquisition via incremental conceptual clustering[J]. Machine Learning, 1987,2(5):139-172.

[41] Kohonen T. Self-organizing maps[M].Second edition. Berlin: Spring Verlag, 1997.

[42] Ester M, Krigel H, Sander P J, Xu X A . density-based algorithm for discovering

clustering in large spatial databases with Noise[J]. Proc．2nd Int. Conf. On Knowledge Discovery and Data Mining, Portland, 1996, 2(26):-21-31.

[43] Gibson D, Kleinberg J M, Raghavan P. Clustering categorical data：An approach based on dynamical systems[J]．Proceedings of the 24th International Conference on Very Large Data Bases,1998,311—322.

[44] L. A. Zadeh. Classification and Clustering[M]. New York.: Academic Press, 1977.

[45] Jain A K, Murty M N. Flyrm P J. Data clustering[J]. A review．ACM Computing Surveys, 1999,31(3):264-323.

[46] 王丽．图论在算法设计中的应用[M].西安:西安电子科技大学,2010.

[47] M. Ramze, Rezaee, B.P.F.Lelieveldt, J.H.C.Reiber. A new cluster validity index for the fuzzy c-mean[J]. Pattern Recognition Letter, 1998,19(6):237-246.

[48] Zhang X, Jiao L, Liu F, et al. Spectral clustering ensemble applied to SAR image segmentation [J]. IEEE Transactions on Geosciences and Remote Sensing, 2008, 46(7):2126-2136.

[49] Fiedler M. A property of eigenvectors of non-negative symmetric matrices and its application to graph theory[J].Czech Math J, 1975,25(100):619-633.

[50] 贾建华著.谱聚类集成算法研究[M]．天津：天津大学出版社，2011.

[51] 王雪松，张晓丽，程玉虎，李立晶.一种基于谱聚类的聚类核半监督支持向量机[J].中国矿业大学学报.2010,39(1):886-890.

[52] Wu Z，Leahy R . An optimal graph theoretic approach to data clustering：theory and its application to image segmentation[J]．IEEE Trans on PAMI, 1993, 15(11):1101-1113.

[53] Hagen L，Kahng A B. New spectral methods for ratio cut portioning and clustering[J]．IEEE Trans．Computer Aided Design,1992, 11(9):1074-1085.

[54] Sarkar S, Soundararajan P. Supervised learning of large perceptual organization：Graph spectral partitioning and learning automata[J]．IEEE Transaction on Patten Analysis and Machine Intelligence, 2000, 22(5):504-520.

[55] Ding C，He X，Zha H．et al spectral Min-Max cut for Graph Partitioning and Data

clustering[c].Proc. of the IEEE Intl Conf. on Data Mining. 2001

[56] Meila M，Xu L．Multiway cuts and spectral clustering[J]. Washington Tech Report 2003.

[57] 孙继广．矩阵扰动分析[M].北京：科学出版社,2001.

[58] Bezdek J C. Pattern recognition with fuzzy objective function algorithm [M]. New York: Plenum Press,1981.

[59] Krishnapuram R, Keller J M. A possibilistic approach to clustering [J]. IEEE Trans on Fuzzy Systems, 1993, 1(1): 98-110.

攻读硕士期间发表的论文

[1] 张亚平，杨明. 一种基于密度敏感的自适应谱聚类算法[J].数学的实践与认识，2013.10.(10):150-156.

[2] 张亚平. 具有反馈顾客的可修M/M/1排队模型及应用[J].商丘师范学院学报,2013.12(12):14-18.

[3] 《一种基于谱聚类的自适应IPCM算法》数学的实践与认识（终审中）

致 谢

三年的时光依稀还在眼前，几多怀想，几多留恋，奋斗与辛劳已化作丝丝回忆，甜美与欢笑也将尘埃落定。借此毕业论文完成之际，我谨向所有关心、爱护、帮助我的人们表达我最诚挚的感谢和最美好的祝福。

首先感谢我的导师杨明老师，杨老师渊博的知识、严谨的治学态度、精益求精的工作作风、诲人不倦的高尚师德、朴实无华的、平易近人的人格魅力对我影响深远。特别在学术上点拨式的指引，留给了我广阔的自由探索的空间，使我受益终身。三年来的悉心教诲，永生难忘。在此向杨老师，表达我最深切的谢意与最衷心的祝福。

感谢范艳玲、辛晚霞、赵可、张艳、张佩宇、刘旭等同学三年的并肩作战，有了你们我的三年生活才多姿多彩。

同时，感谢父母的养育以及家人的支持。是他们无私的关爱，给予我生活的动力和前进的勇气。而今即将步入社会，希望在今后的人生中能够做出更正值得他们欣慰的业绩，用自己的努力让他们生活得更好。

最后，感谢评审学者及答辩委员会的专家们在百忙之中抽出时间对我的论文审阅和指导。