计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2017,26(10):150−155 [doi: 10.15888/j.cnki.csa.005988]
©中国科学院软件研究所版权所有.
E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

# 基于改进高斯核度量和 KPCA 的数据聚类新方法①

余文利 [1], 余建军 [1], 方建文 [2]

[1](衢州职业技术学院 信息工程学院, 衢州 324000)
[2](衢州学院 电气与信息工程学院, 衢州 324000)

**摘　要**: 大多数超椭球聚类 (hyper-ellipsoidal clustering, HEC) 算法都使用马氏距离作为距离度量, 已经证明在该条件下划分聚类的代价函数是常量, 导致 HEC 无法实现椭球聚类. 本文说明了使用改进高斯核的 HEC 算法可以解释为寻找体积和密度都紧凑的椭球分簇, 并提出了一种实用 HEC 算法-K-HEC, 该算法能够有效地处理椭球形、不同大小和不同密度的分簇. 为实现更复杂形状数据集的聚类, 使用定义在核特征空间的椭球来改进 K-HEC 算法的能力, 提出了 EK-HEC 算法. 仿真实验证明所提出算法在聚类结果和性能上均优于 K-means 算法、模糊 C-means 算法、GMM-EM 算法和基于最小体积椭球 (minimum-volume ellipsoids, MVE) 的马氏 HEC 算法, 从而证明了本文算法的可行性和有效性.

**关键词**: 数据聚类; 超椭球聚类; 最小体积椭球; 核主成分分析; 高斯核

引用格式: 余文利,余建军,方建文.基于改进高斯核度量和 KPCA 的数据聚类新方法.计算机系统应用,2017,26(10):150–155. http://www.c-s-a.org.cn/1003-3254/5988.html

## Novel Data Clustering Method Based on A Modified Gaussian Kernel Metric and Kernel PCA

YU Wen-Li[1], YU Jian-Jun[1], FANG Jian-Wen[2]

[1](College of Information Engineering, Quzhou College of Technology, Quzhou 324000, China)
[2](College of Electrical and Information Engineering, Quzhou University, Quzhou 324000, China)

**Abstract**: Most hyper-ellipsoidal clustering(HEC) algorithms use the Mahalanobis distance as a distance metric. It has been proven that HEC, under this condition, cannot be realized since the cost function of partitional clustering is a constant. We demonstrate that HEC with a modified Gaussian kernel metric can be interpreted as a problem of finding condensed ellipsoidal clusters(with respect to the volumes and densities of the clusters) and propose a practical HEC algorithm named K-HEC that is able to efficiently handle clusters that are ellipsoidal in shape and that are of different size and density. We then try to refine the K-HEC algorithm by utilizing ellipsoids defined on the kernel feature space to deal with more complex-shaped clusters. Simulation experiments demonstrate the proposed methods have a significant improvement in the clustering results and performance over K-means algorithm, fuzzy C-means algorithm, GMM-EM algorithm and HEC algorithm based on minimum-volume ellipsoids using Mahalanobis distance.

**Key words**: data clustering; hyper-ellipsoidal clustering; minimum-volume ellipsoids; kernel PCA; Gaussian kernel

聚类作为一种重要的数据分析手段, 是机器学习、模式识别、计算机视觉和数据挖掘等领域的研究热点[1]. 聚类分析就是把对象按照性质上的亲疏程度分多个类或簇, 使得簇内的数据具有较高的相似度, 簇间的数据具有较高的相异度[2]. 尽管许多研究者在不断努力, 但目前仍没有一种能够处理所有聚类问题的最优算法, 聚类仍然是一个困难和具有挑战性的问题.

传统的聚类算法如 K-means、GMM-EM、模糊

① 收稿时间: 2017-01-11; 采用时间: 2017-02-15



© 中国科学院软件研究所 http://www.c-s-a.org.cn

C-means(FCM) 等, 都是基于最小化簇内样本点的欧氏距离和的通用聚类准则, 基于欧氏度量的聚类算法倾向于将样本点划分到相同大小、相同密度和球形的分簇中, 这些算法无法完成大而细长的分簇的划分, 而现实世界的数据常常以混合高斯分布的形式呈现, 如椭球形或其他复杂形状. 为解决上述问题, 人们提出了各种超椭球聚类算法 (hyper-ellipsoidal clustering, HEC)[3-10], 这些算法通常使用马氏距离作为距离度量来建立椭球分簇. 现有的 HEC 算法主要存在以下问题: 1) 过高的计算复杂度, 导致在马氏距离中直接计算协方差矩阵非常困难; 2) 当分簇包含少量样本点时, 协方差矩阵可能是奇异的. 为了克服以上的不足, 文献[3-5]提出基于改进马氏距离和伪协方差矩阵的 HEC 算法, 但是这些算法的时间复杂度仍然很高. 另一方面, 文献[8-10]通过近似分簇体积, 即找到最小体积椭球 (minimum-volume ellipsoids, MVE), 取代了协方差矩阵的计算, 部分克服了以上的不足. 以上大多数方法都是使用马氏距离作为距离度量, 已经证明直接将马氏距离应用于聚类不能获得椭球聚类[7], 而且划分聚类的代价函数也无法限定分簇的大小和分簇之间的关系. 此外, 因为这些算法在聚类时无需考虑样本集的密度, 所以基于 MVE 的 HEC 算法只有等密度分簇的情况才能工作得很好.

本文的目标是实现椭球聚类并给出改进的实用 HEC 算法的实现, 首先, 提出了使用改进高斯核度量的基于 MVE 的 K-HEC 算法, 该算法能够处理椭球形状、不同大小和密度的分簇. 为了增强 K-HEC 算法的能力, 通过在特征空间中映射椭球, 提出了 EK-HEC 算法, 该算法能够处理非线性和细长结构的分簇. 在模拟数据集和标准评测数据集上的仿真实验表明, 本文算法在聚类结果和性能上与 K-means 算法、模糊 C-means 算法、GMM-EM 算法和马氏 MVE-HEC 算法相比有了很大的提高, 从而验证了本文算法在处理椭球形或复杂形状数据集聚类时的可行性和有效性.

## 1 问题定义

一般划分聚类问题可描述为: 给定 $d$ 维空间的 $n$ 个样本, $x=\left\{x_i\in\mathbb{R}^d\right\}_{i=1}^n$, 划分聚类的目标是确定划分矩阵的一个分配, 即 $P=\{P_{ik}|P_{ik}\in[0,1];i=1,2,\ldots,n;k=1,2,\ldots,C\}$, 通过最小化代价函数 $E_C(\mathrm{P})$ 得到

$$\mathbf{P}=\arg(\mathbf{P})\min E_C(\mathbf{P}) \tag{1}$$

其中 $C$ 是由用户确定的类别数. 如果数据点 $x_i$ 被划分到第 $k$ 个分簇, $P_{ik}$=1, 否则 $P_{ik}$=0. 对于一个特定的划分满足 $\sum_{k=1}^{C}P_{ik}=1$. 则划分聚类的代价函数定义为

$$E_C(\mathbf{P})=\sum_{i=1}^{n}\sum_{k=1}^{C}P_{ik}D(x_i,m_k) \tag{2}$$

其中 $D(x_i, m_k)$ 为输入模式与第 $k$ 个分簇 $m_k$ 的均值向量之间的距离度量.

为了构造椭球分簇, HEC 算法通常采用马氏距离. 然而在该条件下, 划分聚类的代价函数是常量[7]. 为了实现椭球聚类, 本文使用式 (3) 的改进高斯核作为距离度量

$$D(x_i,m_k)=\alpha\cdot(x_i-m_k)^T\mathbf{Q}_k^{-1}(x_i-m_k)+(1-\alpha)\cdot\ln(\det\mathbf{Q}_k) \tag{3}$$

其中 $m_k$ 和 $\mathbf{Q}_k$ 分别是第 $k$ 个分簇的均值向量和协方差矩阵. 变量 $\alpha\in[0,1]$控制着式 (3) 的第 1 项和第 2 项的权重. 式 (3) 的第 1 项表示马氏距离, 第 2 项与由协方差矩阵 $\mathbf{Q}_k$ 表示的第 $k$ 个椭圆分簇的容积成正比. 则聚类代价函数改写为

$$E_C(\mathbf{P})=\sum_{i=1}^{n}\sum_{k=1}^{C}P_{ik}[\alpha\cdot(x_i-m_k)^T\mathbf{Q}_k^{-1}(x_i-m_k)+(1-\alpha)\cdot\ln(\det\mathbf{Q}_k)] \tag{4}$$

代价函数 $E_C(\mathbf{P})$ 达到最优的必要条件为 $\partial E_C(\mathbf{P})/\partial m_k^T=0$, 通过最小化式 (4) 得到划分的分簇中心表示为

$$m_k=\sum_{i=1}^{n}P_{ik}x_i/\sum_{i=1}^{n}P_{ik} \tag{5}$$

引理 1. 给定 $d$ 维空间的 $n$ 个样本, $x=\left\{x_i\in\mathbb{R}^d\right\}_{i=1}^n$, 其中 $m_k$ 表示第 $k$ 个分簇中心, $k$=1, 2…, $C$, $C$ 为分簇数. 协方差矩阵 $\mathbf{Q}_k=\frac{1}{n-1}\sum_{i=1}^{n}(x_i-m_k)(x_i-m_k)^T$ 是可逆的, 则有

$$\sum_{i=1}^{n}[\alpha\cdot(x_i-m_k)^T\mathbf{Q}_k^{-1}(x_i-m_k)+(1-\alpha)\cdot\ln(\det\mathbf{Q}_k)]=\alpha\cdot d\cdot(n-1)+n\cdot(1-\alpha)\cdot\ln(\det\mathbf{Q}_k) \tag{6}$$

证明: 根据文献[7]和[4]中的定理 1, 有 $\sum_{i=1}^{n}\alpha\cdot(x_i-m_k)^T\mathbf{Q}_k^{-1}(x_i-m_k)=\alpha\cdot d\cdot(n-1)$ 和 $\sum_{i=1}^{n}(1-\alpha)\cdot\ln(\det\mathbf{Q}_k)=n\cdot(1-\alpha)\cdot\ln(\det\mathbf{Q}_k)$, 所以有

$$\sum_{i=1}^{n}[\alpha\cdot(x_i-m_k)^T\mathbf{Q}_k^{-1}(x_i-m_k)+(1-\alpha)\cdot\ln(\det\mathbf{Q}_k)]=\alpha\cdot d\cdot(n-1)+n\cdot(1-\alpha)\cdot\ln(\det\mathbf{Q}_k)$$

证毕.



© 中国科学院软件研究所 http://www.c-s-a.org.cn

定理 1. 如果改进高斯核式 (3) 作为式 (2) 的聚类代价函数的距离度量, 则

$$E_C(\mathbf{P}) \cong \sum_{k=1}^{C} n_k \ln(\det \mathbf{Q}_k) \tag{7}$$

其中$n_k = \sum_{i=1}^{n} P_{ik}$为划分到第 $k$ 个分簇的样本数.

证明: 根据引理 1, 有$\sum_{i=1}^{n} P_{ik} D(x_i, m_k) = \alpha \cdot d \cdot (n_k - 1) + n_k \cdot (1-\alpha) \cdot \ln(\det \mathbf{Q}_k)$, 则代价函数可以表示为:

$$\begin{aligned} E_C(\mathbf{P}) &= \sum_{i=1}^{n}\sum_{k=1}^{C} P_{ik} D(x_i, m_k) \\ &= \sum_{k=1}^{C} \alpha \cdot d \cdot (n_k - 1) + \sum_{k=1}^{C} n_k \cdot (1-\alpha) \cdot \ln(\det \mathbf{Q}_k) \\ &= \alpha \cdot d \cdot (n - C) + (1-\alpha) \sum_{k=1}^{C} n_k \ln(\det \mathbf{Q}_k) \end{aligned}$$

其中 $\alpha$、$d$、$n$ 和 $C$ 都是关于变量 $k$ 的常量, 因此有 $E_C(\mathbf{P}) \cong \sum_{k=1}^{C} n_k \ln(\det \mathbf{Q}_k)$.

证毕.

## 2 椭球和复杂形状聚类

本文提出了两种最小化分簇体积权重和的 HEC 算法—K-HEC 算法和 EK-HEC 算法. 其中 K-HEC 算法是使用改进高斯核和 MVE 近似的迭代 HEC 算法, 它将样本划分到指定数量的椭球分簇中; EK-HEC 算法是 K-HEC 算法的扩展, 它通过使用定义在核特征空间的椭球来改善 K-HEC 算法的聚类能力, 以便能处理更复杂形状样本集的聚类.

### 2.1 K-HEC 算法

K-HEC 算法开始于一个初始的聚类, 最终将初始的聚类划分为 $C$ 个分簇, 在此过程中算法迭代查找改进的划分矩阵的分配, 分簇体积的权重和不断缩小, 直到分簇结果没有进一步可能的改进为止.

给定 $d$ 维空间的 $n$ 个样本, $x = \left\{x_i \in \mathbb{R}^d\right\}_{i=1}^{n}$, 计算 MVE 所包含样本可以定义为求最大特征向量问题[9-11], 以上问题难以通过直接求以下优化问题解决.

$$\begin{cases} \min \ln(\det \mathbf{Q}) \\ s.t. \mathbf{Q} = \\ \mathbf{Q}^T > 0 and (x_i - x_C)^T \mathbf{Q}^{-1} (x_i - x_C) \le 1, i = 1, 2 \ldots, n \end{cases} \tag{8}$$

因此, 本文使用三个近似方法来寻找 MVE. 首先, 推导式 (9) 所示的 Löwner-John 椭球体[11]凸优化问题, 该问题可以几何解释为最小化椭球体的体积.

$$\begin{cases} \min \det \mathbf{A}^{-1} \\ s.t. \mathbf{A} > 0 and \|\mathbf{A}x_i - b\| \le 1, i = 1, 2, \ldots n \end{cases} \tag{9}$$

其次, 通过近似计算包含矩阵 $\mathbf{Q}$ 特征向量和的目标函数来求解式 (10) 所示凸优化问题[8,11].

$$\begin{cases} \min \mathrm{Trace}(\mathbf{Q}) \\ s.t. \mathbf{Q} = \\ \mathbf{Q}^T > 0 and (x_i - x_C)^T \mathbf{Q}^{-1} (x_i - x_C) \le 1, i = 1, 2, \ldots n \end{cases} \tag{10}$$

最后, 使用 Khachiyan 的快速近似算法[12]来寻找 MVE.

$$\begin{cases} \mathrm{MVE} = \{x \in \mathbb{R}^d | (x - x_C^*)^T \mathbf{Q}^* (x - x_C^*) \le 1\} \\ where \mathbf{Q}^* = \frac{1}{d} (\mathbf{P}\mathbf{U}^* \mathbf{P}^T - \mathbf{P}u^* (\mathbf{P}u^*)^T)^{-1}, x_C^* = \mathbf{P}u^* \end{cases} \tag{11}$$

其中 $\mathbf{P} = [q_1, q_2, \ldots, q_n] \in \mathbb{R}^d$, $q_i^T = [x_i^T\ 1]$, $i$=1, 2, …, $n$, $u$ 为对偶变量, $\mathbf{U}$=diag($u$). 则 K-HEC 算法描述如算法 1.

算法1. K-HEC算法

输入: $d$维空间的$n$个样本$x = \left\{x_i \in \mathbb{R}^d\right\}_{i=1}^{n}$

输出: 划分矩阵$\mathbf{P}$

步骤1. 确定分簇数$C$, 从样本集中随机选择$C$个样本, 设定为分簇的中心, 记作$m_k$;

步骤2. 首先使用样本与分簇中心$m_k$欧氏距离来确定划分矩阵的初始分配

$$\begin{cases} P_{ik} = 1, if D_{Euc}(x_i, m_k) < D_{Euc}(x_i, m_j), \\ \qquad j = 1, 2, \ldots, C and j \ne k \\ P_{ik} = 0, otherwise \end{cases}$$

步骤3. 计算新的分簇中心和属于该分簇的样本的数量

$$m_k = \frac{\sum_{i=1}^{n} P_{ik} x_i}{\sum_{i=1}^{n} P_{ik}}, n_k = \sum_{i=1}^{n} P_{ik}$$

步骤4. 使用MVE近似算法计算伪协方差矩阵$\mathbf{Q}_k$;

步骤5. 使用式(3)的改进高斯核度量、$m_k$和$\mathbf{Q}_k$确定划分矩阵$\mathbf{P}$的一个新的分配

$$\begin{cases} P_{ik} = 1, if D_{MGK}(x_i, m_k; \mathbf{Q}_k) < D_{MGK}(x_i, m_j; \mathbf{Q}_j) \\ P_{ik} = 0, otherwise \end{cases}$$

步骤6. 如果划分矩阵$\mathbf{P}$没有变化, 则算法停止. 否则重复步骤3到步骤5.

### 2.2 EK-HEC 算法

大部分划分聚类算法在对非线性和细长结构数据集进行聚类时, 不能取得理想的效果, 为此人们提出了谱聚类算法[13]和核方法[14]. 为了在非线性和细长结构数据集上执行聚类, 本文使用了核主成分分析 (Kernel principal component analysis, KPCA). 通过 RBF 核函数, 能够有效地在与基于非线性映射的输入空间相关联的高维特征空间中计算主成分. KPCA 的原理如下所示.

给定样本集$X = \{x_i\}_{i=1}^{n}$, $\Phi$是非线性映射, 且满足$\sum_{i=1}^{n} \Phi(x_i) = 0$, 对应的空间记为 $F$, 则对应的协方差矩阵为$\mathbf{C}^{\Phi} = \frac{1}{n} \sum_{j=1}^{n} \Phi(x_j) \Phi(x_j)^T$, 对 $\mathbf{C}^{\Phi}$进行特征分解得



© 中国科学院软件研究所 http://www.c-s-a.org.cn

$\lambda \boldsymbol{V}=\mathbf{C}^{\Phi}\boldsymbol{V}$, 特征向量 $\boldsymbol{V}$ 可由 $F$ 空间中的样本张成, 记作 $\boldsymbol{V}=\sum_{i=1}^{n}\alpha_i\Phi(x_i)$. 引入核函数 $K_{ij}=\left(\Phi(x_i)\cdot\Phi\left(x_j\right)\right)$, $i,j=1,2,\ldots\ldots,n$, 可以转变为求核矩阵 $\mathbf{K}$ 的特征向量和特征值

$$n\lambda\alpha=\mathbf{K}\alpha \tag{12}$$

假定对应于非 0 特征值的特征向量分别为 $\alpha^1,\alpha^2,\ldots,\alpha^p$, 将 $\boldsymbol{V}^k$ 归一化 ($k$=1, 2, …, $p$), 此时样本 $\Phi(x)$ 在 $\boldsymbol{V}^k$ 上的投影为: $\tilde{x}=(\boldsymbol{V}^k\cdot\Phi(x))=\sum_{i=1}^{n}\alpha_i^kK(x_i,x)$, 其中 $K(x,y)$ 为核函数, 本文使用 RBF 核函数 $K(x,y)=\exp\left(-\|x-y\|^2/2\sigma^2\right)$.

EK-HEC 算法通过在核空间中执行 K-HEC 算法实现复杂形状聚类, 算法的具体描述如算法 2.

算法2. EK-HEC算法

输入: $d$维空间的$n$个样本$x=\left\{x_i\in\mathbb{R}^d\right\}_{i=1}^{n}$

输出: 划分矩阵$\mathbf{P}$

步骤1. 计算核矩阵$\mathbf{K}$, $K_{ij}=\Phi(x_i)\cdot\Phi(x_j)=K(x_i,x_j)$;$i,j=1,2,\ldots,n$;

步骤2. 解式(12), 找到非零的特征值;

步骤3. 使用非零特征值$\alpha^k$($k$=1, 2, …, $p$), 计算样本$\Phi(x)$在特征向量$\boldsymbol{V}^k$上的投影$\tilde{x}=(\boldsymbol{V}^k\cdot\Phi(x))=\sum_{i=1}^{n}\alpha_i^kK(x_i,x)$;

步骤4. 确定分簇数$C$, 在特征空间中随机选择$C$个样本, 将它们设置为分簇的中心, 记作$\tilde{m}_k$;

步骤5. 使用欧氏距离计算样本与分簇中心$\tilde{m}_k$的距离, 确定划分矩阵$\mathbf{P}$的初始分配

$$\begin{cases} P_{ik}=1, if\, D_{Euc}(\tilde{x}_i,\tilde{m}_k)<D_{Euc}(\tilde{x}_i,\tilde{m}_j), \\ \qquad j=1,2,\ldots C and\, j\neq k \\ P_{ik}=0, otherwise \end{cases}$$

步骤6. 计算新的分簇中心和属于新分簇的样本数

$\tilde{m}_k=\sum_{i=1}^{n}P_{ik}\tilde{x}_i/\sum_{i=1}^{n}P_{ik}, n_k=\sum_{i=1}^{n}P_{ik}$

步骤7. 使用MVE近似算法计算伪协方差矩阵$\tilde{\mathbf{Q}}_k$;

步骤8. 使用式(3)的改进高斯核度量、$\tilde{m}_k$和$\tilde{\mathbf{Q}}_k$确定划分矩阵$\mathbf{P}$的新的分配

$$\begin{cases} P_{ik}=1, if\, D_{Euc}(\tilde{x}_i,\tilde{m}_k;\tilde{\mathbf{Q}}_k)<D_{Euc}(\tilde{x}_i,\tilde{m}_j;\tilde{\mathbf{Q}}_j), \\ \qquad j=1,2,\ldots C and\, j\neq k \\ P_{ik}=0, otherwise \end{cases}$$

步骤9. 如果划分矩阵$\mathbf{P}$没有变化, 则算法终止; 否则重复步骤6至步骤8.

## 3 仿真实验

### 3.1 实验描述

为验证 K-HEC 算法和 EK-HEC 算法的有效性, 在模拟数据集和基准评测数据集上进行了实验. K-HEC 算法和 EK-HEC 算法是在 Matlab 上使用 CVX[15]和 LMI 工具箱编程实现, 在 Intel(R) Xeon® CPU W5590@ line-height:15.5pt3.33GHZ 的微机 Windows XP 环境下运行. 在性能评估时, 使用误分类率 (Misclassification rate, MCR) 和归一化互信息 (Normalized mutual information, NMI) 作为评价指标, 分别定义如下

$$\mathrm{MCR}=\frac{\text{误分类的样本数}}{\text{总的样本数}}\times 100\% \tag{13}$$

$$\mathrm{NMI}(X,Y)=\frac{I(X,Y)}{[H(X)+H(Y)]} \tag{14}$$

其中 $X$、$Y$ 是两个随机变量, $I(X,Y)$ 是互信息, $H(X)$ 和 $H(Y)$ 是 $X$ 和 $Y$ 的熵.

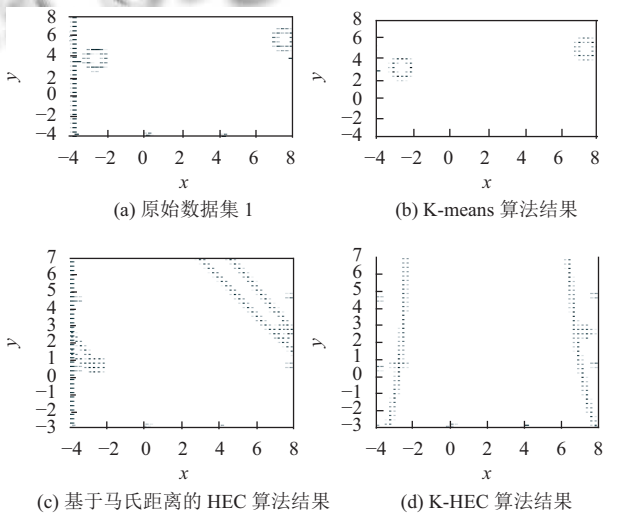
### 3.2 实验结果与分析

#### 3.2.1 模拟数据集

为了说明本文所提出算法的有效性, 在实验中使用了 2 个模拟数据集 (具体描述如表 1 所示). 模拟数据集 1 用以验证 K-HEC 算法对于不同大小、不同密度和椭圆形分簇的聚类能力, 该数据集包含一个圆形的分簇和一人细长椭圆形的分簇. K-HEC 算法在模拟数据集 1 上使用式 (9)-式 (11) 三种不同的 MVE 近似方法所提到的聚类结果是一样的, 因此忽略式 (9) 和式 (10) 的方法, 使用式 (11) 方法的 K-HEC 算法在数据集 1 上的聚类结果如图 1 所示.

表 1 模拟数据集

| 数据集 | 数据集1 | | 数据集2 | |
|---|---|---|---|---|
| 描述 | 球-椭圆形 | | 高斯-香蕉形 | |
| 维数 | 2 | | 2 | |
| 分簇数 | 2 | | 2 | |
| 记录数 | 14 | | 128 | |
| | 4 | 10 | 100 | 28 |



(a) 原始数据集 1 (b) K-means 算法结果

(c) 基于马氏距离的 HEC 算法结果 (d) K-HEC 算法结果

图 1 不同算法在模拟数据集 1 上的聚类结果

从图 1(b) 可以看出, K-means 算法在模拟数据集



© 中国科学院软件研究所 http://www.c-s-a.org.cn

1 上不能得到正确的聚类; 从 (c) 可以看出, 马氏 HEC 算法虽然将样本划分为不同大小的两个椭圆分簇, 但当两个分簇距离较近时聚类结果也不准确; (d) 表明本文提出的 K-HEC 算法通过调整 $\alpha$ 的值来控制改进高斯核的第 1 项和第 2 项的权重, 从而最小化分簇体积权重和, 使得聚类后分簇的紧凑性和密度达到最大.

模拟数据集 2 包含一个高斯分布分簇和一个香蕉形分簇, 用于验证 EK-HEC 算法的有效性, 该算法专门设计用于复杂几何形状样本集的聚类. K-means 算法、HEC 算法、K-HEC 算法和 EK-HEC 算法在模拟数据集 2 上的聚类结果如图 2(b)~(f) 所示.

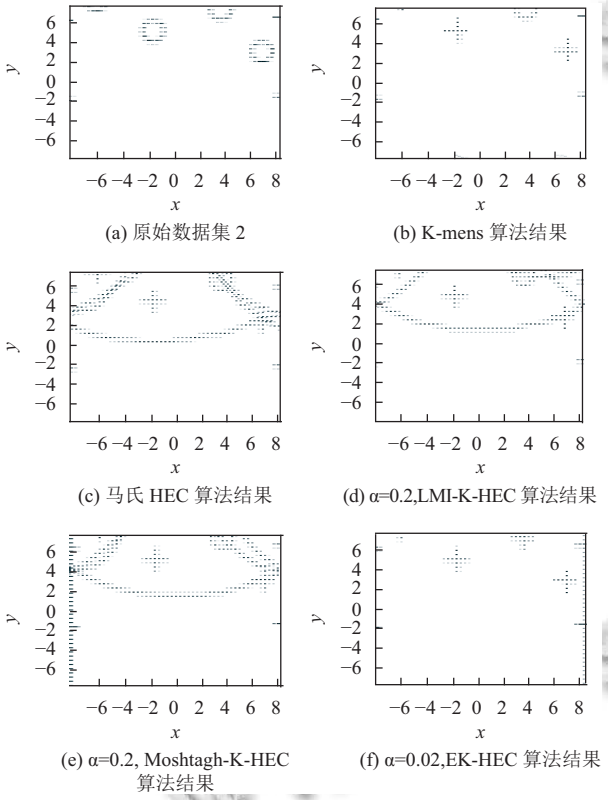



图 2 不同算法在模拟数据集 2 上的聚类结果

从图 2 可以看出, K-means 算法、马氏 HEC 算法以及 K-HEC 算法有相似的聚类结果. 可以注意到, 虽然聚类结果相似, 但是由每个算法所确定的聚类的决策边界仍有很大的不同. 与其他算法相比, 基于 Moshtagh 方法的 MVE 近似的 Moshtagh-K-HEC 算法建立了清晰的决策边界, 而马氏 HEC 算法和 LMI-K-HEC 算法有重叠的决策边界. EK-HEC 算法按照预期找到了正确的分簇, 仅有一个样本错分. 图 3 描述了 EK-HEC 算法在模拟数据集 2 上的聚类结果, 其中 (a) 和 (b) 分别描述了在算法在输入空间和特征空间的聚类结果, 映射到特征空间的样本得到很好的分离, 并且能够通过椭圆的决策边界实现聚类.

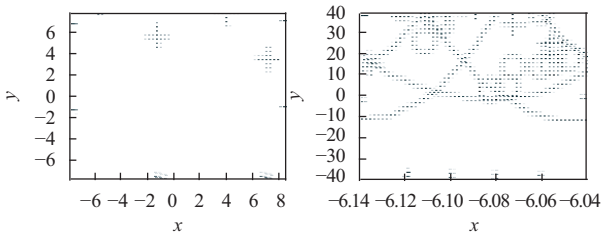



图 3 EK-HEC 算法在模拟数据集 2 上的聚类结果

### 3.2.2 基准评测数据集

为了评估本文提出算法的性能, 在来自 UCI 的 3 个基准评测数据集 (具体描述见表 2) 上与 K-means 算法、模糊 C-means 算法、GMM-EM 算法和马氏距离 MVE-HEC 算法进行了比较实验. 6 种算法在 MCR 和 NMI 两个评判准则上的比较结果如图 4 所示, 由图 4 可知, K-HEC 算法在 MCR 和 NMI 两个指标上均优于 K-means、模糊 C-means、GMM-EM 和 HEC 算法, EK-HEC 算法与其他算法相比有更好或类似的聚类性能.

表 2 来自 UCI 的基准评测数据集

| 数据集 | | Iris | Wine | Glass |
|---|---|---|---|---|
| 属性数 | | 4 | 13 | 11 |
| 类别数 | | 3 | 3 | 7 |
| 记录数 | | 150 | 178 | 214 |
| $\alpha$的取值 | K-HEC | 0.3 | 0.6 | 0.95 |
| | EK-HEC | 0.5 | 0.45 | 0.4 |

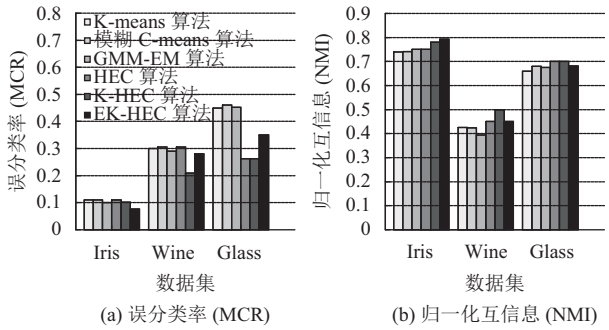
## 4 结语

本文将基于 MVE 的 HEC 算法与改进的高斯核相结合, 提出了 K-HEC 算法, 通过应用定义在核空间的椭圆聚类, 增强了 K-HEC 算法聚类能力, 提出了 EK-HEC 算法. K-HEC 算法能够处理不同大小、不同密度和椭球形壮的分簇, EK-HEC 算法能够处理非线性和细长结构的复杂几何形状的分簇. 在模拟数据集和 UCI 基准评测数据集上的仿真实验表明, K-HEC 算法能够通过建立紧凑的分类边界有效地分离各分簇, EK-HEC 算法在非线性和细长结构的数据集上完成了正确的聚类. 本文算法无论在聚类能力和性能方面均优于 K-means、



© 中国科学院软件研究所 http://www.c-s-a.org.cn

模糊 C-means、GMM-EM 和 HEC 算法, 从而验证了本文算法的可行性和有效性.




(a) 误分类率 (MCR) (b) 归一化互信息 (NMI)

图 4 基准评测数据集上的聚类性能比较

## 参考文献


1 Duda RO, Hart PE, Stork DG. Pattern Classification. New York: John Wiley & Sons, 2012.

2 Nagpal A, Jatain A, Gaur D. Review based on data clustering algorithms. Proc. of the 2013 IEEE Conference on Information & Communication Technologies (ICT). JeJu Island, Korea. 2013. 298–303.

3 Mao JC, Jain AK. A self-organizing network for hyperellipsoidal clustering (HEC). IEEE Trans. on Neural Networks, 1996, 7(1): 16–29. [doi: 10.1109/72.478389]

4 Wang S, Ma F, Shi W, *et al*. The hyperellipsoidal clustering using genetic algorithm. Proc. of the 1997 IEEE International Conference on Intelligent Processing Systems. Beijing, China. 1997, 1. 592–596.

5 Ichihashi H, Ohue M, Miyoshi T. Fuzzy C-means clustering algorithm with pseudo Mahalanobis distances. Proc. of the 3rd Asian Fuzzy Systems Symposium. Changwon, Korea. 1998. 148–152.

6 Moshtaghi M, Rajasegarar S, Leckie C, *et al*. An efficient hyperellipsoidal clustering algorithm for resource-constrained environments. Pattern Recognition, 2011, 44(9): 2197–2209. [doi: 10.1016/j.patcog.2011.03.007]

7 Wang S, Xia SW, Mao JC, *et al*. Comments on “a self-organizing network for hyperellipsoidal clustering (HEC)”. IEEE Trans. on Neural Networks, 1997, 8(6): 1561–1563. [doi: 10.1109/72.641479]

8 Lee HS, Park JY, Park DH. Hyper-ellipsoidal clustering algorithm using linear matrix inequality. Journal of Korean Institute of Intelligent Systems, 2002, 12(4): 300–305. [doi: 10.5391/JKIIS.2002.12.4.300]

9 Kumar M, Orlin JB. Scale-invariant clustering with minimum volume ellipsoids. Computer & Operations Research, 2008, 35(4): 1017–1029.

10 Shioda R, Tunçel L. Clustering via minimum volume ellipsoids. Computational Optimization and Applications, 2007, 37(3): 247–295. [doi: 10.1007/s10589-007-9024-1]

11 Boyd S, Vandenberghe L. Convex optimization. Cambridge, UK: Cambridge University Press, 2004.

12 Todd MJ, Yildirim EA. On Khachiyan’s algorithm for the computation of minimum-volume enclosing ellipsoids. Discrete Applied Mathematics, 2007, 155(13): 1731–1744. [doi: 10.1016/j.dam.2007.02.013]

13 Cao JZ, Chen P, Zheng Y, *et al*. A max-flow-based similarity measure for spectral clustering. ETRI Journal, 2013, 35(2): 311–320. [doi: 10.4218/etrij.13.0112.0520]

14 Shawe-Taylor J, Cristianini N. Kernel methods for pattern analysis. Cambridge, UK: Cambridge University Press, 2004.

15 CVX Research, Inc. CVX: Matlab software for disciplined convex programming, version 2.0. http://cvxr.com/cvx. [2013-04].




© 中国科学院软件研究所 http://www.c-s-a.org.cn