基于最大中心间隔的缩放型 η-极大熵聚类算法*

陈爱国[a]，蒋亦樟[b]，钱鹏江[b]
（江南大学 a. 物联网工程学院；b. 数字媒体学院，江苏 无锡 214122）

摘　要：为了调控数据之间的差异性，一般化的处理方式是对数据简单地进行按比例缩放，而此类做法本身对于数据的信息是不存在任何破坏的。但在进行聚类分析时，大部分算法对于按比缩放的数据都是很敏感的，其中较典型的算法有极大熵聚类（MEC）算法。大量的实验表明，当缩放尺度位于 10^{-3} 数量级以下时，极大熵聚类算法已经失效，通过该算法得到的聚类中心趋于一致。为了解决上述问题，在 MEC 算法的基础上引入最大中心间隔项与缩放因子 η，构造出了全新的目标函数，称为 η 型最大中心间隔极大熵聚类（η-MCS-MEC）算法。该算法通过调控中心点间的距离使之达到最大，并有效利用缩放因子 η 对各类划分进行调控，从而避免了聚类中心趋于一致。通过在模拟数据集以及 UCI 仿真数据集上的实验，结果均显示出算法对变化的数据不再敏感而具有鲁棒性。

关键词：最大中心间隔；数据缩放；极大熵聚类；中心一致
中图分类号：TP391.41　　**文献标志码**：A　　**文章编号**：1001-3695(2013)01-0103-04
doi:10.3969/j.issn.1001-3695.2013.01.024

Maximum center interval and scaling type of η-maximum entropy clustering

CHEN Ai-guo[a], JIANG Yi-zhang[b], QIAN Peng-jiang[b]
(*a. School of Internet of Thing, b. School of Digital Media, Jiangnan University, Wuxi Jiangsu* 214122, *China*)

Abstract: In order to control the difference between data, the general way is to scale the data proportionally, and such practices itself do not have any damage to the information of data. However, most algorithms are very sensitive to the scaling data in the cluster analysis and one of the typical algorithms is MEC algorithm. A lot of experiments show that MEC algorithm has failed when the zoom level locating below 10^{-3} orders of magnitude, and the cluster centers obtained by the algorithm are likely to have consistency clustering. To solve the above problems, this paper introduced the largest center of interval and the scaling factor η to restructure a new objective function, which called the maximum center interval maximum entropy clustering (η-MCS-MEC) algorithm. This algorithm achieved the maximum by adjusting the distance between the center points and controled the division of each cluster by using the scaling factor η effectively, and which avoided the agreement of the clustering centers. Numerical experiments conducting on the UCI standard data sets and artificial data sets show that the proposed algorithm is not sensitive to the changing data and has better robustness.

Key words: maximum center interval; data scaling; maximum entropy clustering(MEC); same center

0　引言

在数据挖掘及模式识别领域内，有关聚类技术的分析与讨论总是不间断的。而在各种的聚类技术中又以基于划分的聚类算法最为常用。在这些算法中比较著名的有基于模糊理论的模糊聚类算法，最具代表性的有 FCM 算法[1]，该类技术被广泛应用于模式识别的各个领域[2~4]，以此算法为基础又出现了相关的改进算法如 AFCM[5]、PFCM[6] 和 CFCM[7] 等算法。无论经过了何种改进，该类算法都始终以模糊划分作为其基础。在 1995 年，Li 等人[8] 创造性地在 C-均值算法的基础上引入了熵的概念，并通过将极大熵应用于最小化均方误差的思想，构造出了新的聚类方法，即极大熵聚类（MEC）算法。这一方法比以往的聚类技术具有更为简洁的数学表达和明确的物理含义，引发了许多研究人员的兴趣。后来有人以此为基础进行了改进，得到了如 MECA[9]、FBACN[10] 及 RMEC[11] 等算法。

上述改进算法在一定的领域内有着更好的适用能力，但其本质仍然是以最初的 MEC 算法为基础。在经过一系列的研究与分析之后，发现 MEC 算法在处理微量级或缩放处理过的数据时，由于其划分的计算策略问题，导致该算法对变化的数据非常敏感，并在数量级达到一定的阶级时，传统 MEC 算法得到的聚类中心会发生中心一致的现象，这直接导致了聚类效果的明显恶化，特别在数量级处于毫米级以下时，该算法将直接失效。此问题的存在严重影响了极大熵聚类的适用领域以及面对变化数据的鲁棒性。

为了解决上述问题，以使极大熵聚类算法更具适用能力，

收稿日期：2012-05-21；**修回日期**：2012-06-24　　**基金项目**：国家自然科学基金资助项目(90820002)；江苏省自然科学基金资助项目(BK2009067)

作者简介：陈爱国(1975-)，男，江苏靖江人，讲师，主要研究方向为人工智能与模式识别(agchen_1975@163.com)；蒋亦樟(1988-)，男，江苏无锡人，博士研究生，主要研究方向为模式识别、智能计算及应用；钱鹏江(1979-)，男，江苏泰州人，副教授，博士，主要研究方向为数据挖掘、智能计算及应用.

并提高了该算法面对变化数据时的鲁棒性。本文通过引入最大中心间隔项以及缩放因子 η,构造出了新的聚类算法,称之为 η 型最大中心间隔极大熵聚类(η-MCS-MEC)算法。该算法利用缩放因子调控数据之间的差异程度,再根据最大中心间隔项拉大类与类之间的间隔,使之尽量远离,避免了中心一致现象的产生。通过这两项的有效控制,使得 η-MCS-MEC 算法在面对变化的数据时,较传统的 MEC 算法不再敏感。

1 极大熵聚类(MEC)算法

在传统聚类手段中,一个重要的技术是引入熵概念的聚类方法。通过引入熵得到的算法比其他的聚类技术,如 FCM[1] 以及 PCM[12,13] 等算法,有着更加简洁的数学形式以及更为合理的物理解释。

1.1 MEC 算法原理

MEC 算法的表达形式很多,但最为经典的仍属文献[5]中提及的目标函数式,其具体表达式为

$$J_{\mathrm{MEC}}(U,V)=\sum_{i=1}^{C}\sum_{j=1}^{N}\mu_{ij}\|x_j-v_i\|^2+\gamma\sum_{i=1}^{C}\sum_{j=1}^{N}\mu_{ij}\ln\mu_{ij} \tag{1}$$

其中:$\mu_{ij}\in[0,1]$,$1\leqslant i\leqslant C$,$1\leqslant j\leqslant N$,$\sum_{i=1}^{C}\mu_{ij}=1$。

通过上述限制条件利用拉氏法则进行推导,可得目标函数式(1)取得极值的必要条件为

$$\mu_{ij}=\frac{\exp\left(-\frac{\|x_j-v_i\|^2}{\gamma}\right)}{\sum_{k=1}^{C}\exp\left(-\frac{\|x_j-v_k\|^2}{\gamma}\right)}\quad i=1,2,\cdots,C;j=1,2,\cdots,N \tag{2}$$

$$v_i=\frac{\sum_{j=1}^{N}\mu_{ij}^m x_j}{\sum_{j=1}^{N}\mu_{ij}^m}\quad i=1,2,\cdots,C \tag{3}$$

通过上述推导,MEC 算法可归结为以下步骤:

a)初始化聚类数 C,最大迭代次数 max_iterative,迭代阈值 ε,初始迭代次数 $n=1$ 以及初始划分矩阵 $U(n)$。

b)通过式(3)更新得到新的中心坐标 $V(n)$。

c)通过式(2)以及步骤 b)得到新的 $V(n)$,更新得到新的划分矩阵 $U(n+1)$。

d)当 $\|U(n+1)-U(n)\|_{\mathrm{Frobenius}}<\varepsilon$ 或 n 达到最大迭代次数max_iterative 时,算法终止,否则跳回步骤 b)进行求解直至算法满足终止条件为止。

1.2 中心一致性问题

根据上节对于 MEC 算法原理的分析,观察式(2)和(3),可以得到如下的重要结论。

结论 1 首先对式(2)进行以下的变型转换:

$$\mu_{ij}=\frac{1}{\sum_{k=1}^{C}\exp\left(\frac{\|x_j-v_i\|^2-\|x_j-v_k\|^2}{\gamma}\right)}$$
$$i=1,2,\cdots,C;j=1,2,\cdots,N \tag{4}$$

分析式(4)可知当数据 $X=\{x_1,\cdots,x_N\}$ 缩放 N 倍得到的 $Z=\{z_1,\cdots,z_N\}$,其满足 $Z=N\times X$。因式(4)的问题求解利用了 e 函数,所以经数据缩放后所得的划分矩阵的变化与缩放倍数 N 之间并不呈现线性关系,其具体的变化规律满足函数 e^{-N^2} 的分布,如图 1 所示。

通过图 1 可以发现:当 N 缩放倍数在 1e-1 数量级以下时,e^{-N^2} 趋于 1;当 N 缩放倍数在 10 以上时,e^{-N^2} 则趋于 0。这就生产了两个极端现象:a)当 $N<1e-1$,则所求得的划分矩阵中的元素都将趋于 1,导致算法得到一致性的划分最终产生中心点重合的现象,使得聚类精度大大降低,造成 MEC 算法失效;b)当 $N>10$ 时,则 $\exp(-\gamma^{-1}\|x_j-v_i\|^2)$ 趋于 0,式(4)的变化则无法满足,观察式(2),由于分母亦趋于 0,因此整个划分矩阵将趋于无穷大,此时 MEC 算法失效。

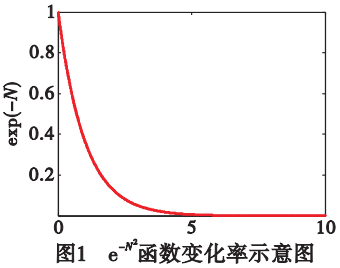


图1 e^{-N^2}函数变化率示意图

经过结论 1 的分析可知,无论是对数据进行缩小还是放大的操作,都将导致 MEC 算法的失效,这证明了 MEC 算法对于数据的缩放是非常敏感的,不具鲁棒性。

2 η 型最大中心间隔极大熵聚类(η-MCS-MEC)算法

为了有效地解决 1.2 节提到的中心一致性问题,本文对 MEC 算法的目标函数式进行了改造,通过引入缩放因子 η,有效地调控由于数据缩放造成的划分矩阵与缩放倍数 N 之间的非线性关系,表达形式如下:

$$p=\eta\sum_{i=1}^{C}\sum_{j=1}^{N}\mu_{ij}\|x_j-v_i\|^2 \tag{5}$$

同时,为了更加有效地避免中心点重合现象,本文又提出了最大中心间隔项:

$$q=\frac{\lambda}{C-1}\sum_{i=1}^{C}\sum_{k=1,k\neq i}^{C}\|v_i-v_k\|^2 \tag{6}$$

利用上述两项,最终得到新的 η 型最大中心间隔极大熵聚类算法目标函数式:

$$J_{\eta\text{-MCS-MEC}}=\eta\left(\sum_{i=1}^{C}\sum_{j=1}^{N}\mu_{ij}\|x_j-v_i\|^2-\frac{\lambda}{C-1}\sum_{i=1}^{C}\sum_{k=1,k\neq i}^{C}\|v_i-v_k\|^2\right)+\gamma\sum_{i=1}^{C}\sum_{j=1}^{N}\mu_{ij}\ln\mu_{ij} \tag{7}$$

$$\text{s.t.}\quad \mu_{ij}\in[0,1],1\leqslant i\leqslant C,1\leqslant j\leqslant N,\sum_{i=1}^{C}\mu_{ij}=1$$

其中:C 表示聚类数目;N 为总体样本的个数;λ 为最大间隔调控因子;η 为缩放因子;v_i 表示第 i 类的中心坐标;x_j 为第 j 个样本点;μ_{ij}表征了第 j 个样本点隶属于第 i 类的隶属度;γ 为隶属度控制项。

2.1 参数优化

为了解决目标函数式(7)的极值问题,本文给出如下定理。

定理 1 式(7)达到最优取得极小值时,其对应的必要条件为

$$\mu_{ij}=\frac{\exp\left(-\eta\frac{\|x_j-v_i\|^2}{\gamma}\right)}{\sum_{k=1}^{C}\exp\left(-\eta\frac{\|x_j-v_k\|^2}{\gamma}\right)} \tag{8}$$

$$v_i=\frac{\sum_{j=1}^{N}\mu_{ij}^m x_j-\frac{\lambda}{C-1}\sum_{k=1,k\neq i}^{C}v_k}{\sum_{j=1}^{N}\mu_{ij}^m-\lambda} \tag{9}$$

其中:$i=1,2,\cdots,C;j=1,2,\cdots,N;\sum_{i=1}^{C}\mu_{ij}=1$。

证明 仿照文献[5]以及 FCM 算法等最优问题的求解策略,得到式(7)在约束条件下的拉格朗日函数为

$$J'(U,V)=\eta(\sum_{i=1}^{C}\sum_{j=1}^{N}\mu_{ij}\|x_j-v_i\|^2-\frac{\lambda}{C-1}\sum_{i=1}^{C}\sum_{k=1,k\neq i}^{C}\|v_i-v_k\|^2)+\gamma\sum_{i=1}^{C}\sum_{j=1}^{N}\mu_{ij}\ln\mu_{ij}+\sum_{j=1}^{N}\alpha_j\sum_{i=1}^{C}(1-\mu_{ij}) \tag{10}$$

其中,α_j 为拉格朗日乘数。因此通过转换,式(7)的极值问题变成了式(10)的极值问题。对于式(10)而言,其取得极值的必要条件则是 J' 分别对参数 U、α、V 的偏导数为 0 时的最优解所对应的函数解。

首先,通过 $\frac{\partial J'}{\partial \alpha_j}=0$,可得

$$\sum_{i=1}^{C}\mu_{ij}=1 \tag{11}$$

另根据 $\frac{\partial J'}{\partial \mu_{ij}}=0$,可知:

$$\mu_{ij}=\exp(\frac{-\eta\|x_j-v_i\|^2-\gamma}{\gamma})\times\exp(\frac{\alpha_j}{\gamma}) \tag{12}$$

将式(12)代入式(11),可得

$$\exp(\frac{\alpha_j}{\gamma})=\frac{1}{\sum_{k=1}^{C}\exp(\frac{-\eta\|x_j-v_k\|^2-\gamma}{\gamma})} \tag{13}$$

再将式(13)代回到式(12)中,最终可求得 μ_{ij} 的优化表达式,进而式(8)得证。参照上述求解的思想,通过 $\frac{\partial J'}{\partial v_i}=0$,可以求得

$$v_i=\frac{\sum_{j=1}^{N}\mu_{ij}^m x_j-\frac{\lambda}{C-1}\sum_{k=1,k\neq i}^{C}v_k}{\sum_{j=1}^{N}\mu_{ij}^m-\lambda}$$

进而式(9)得证。根据条件极值的理论,当 U、V 取得最优解时,则函数 J' 取得极小值,从而目标函数式(7)也相应地求得最小值。综上所述,定理 1 得证。

2.2　本文算法描述

根据定理 1 的推导,可得 η 型最大中心间隔极大熵聚类算法(η-MCS-MEC)的具体执行步骤如下:

a)初始化聚类数 $C(2<C<N)$,设置最大迭代次数 max_iterative,迭代阈值 ε,初始迭代次数 $n=1$,初始划分矩阵 $U(n)$,初始中心矩阵 $V(n)$,缩放因子 η(一般取 $\frac{1}{N}$)以及隶属度控制参数 γ。

b)通过式(9)更新得到新的中心坐标 $V(n+1)$。

c)通过式(8)以及步骤 b)得到的新 $V(n+1)$,更新得到新的划分矩阵 $U(n+1)$。

d)当 $\|U(n+1)-U(n)\|_{\text{Frobenius}}<\varepsilon$ 或 n 达到最大迭代次数 max_iterative 时,算法终止,否则跳回步骤 b)进行求解直至算法满足终止条件为止。

当算法最终终止时,所求得的划分矩阵 U 以及最终的聚类中心 V 均为最优解。

3　实验

为了验证本文算法的有效性,本章将针对缩放型的模拟数据以及 UCI 数据库中的真实数据,通过分析和比对 MEC 算法与 η-MCS-MEC 在此类数据集上的优劣,证明了本文算法较之传统算法对数据的变化更具鲁棒性。

3.1　运行环境

本文的实验部分均采用以下的硬件配置与编程环境,具体情况如表 1 所示。

3.2　评价指标

为了对本文算法以及传统 MEC 算法有一个直观的评价,本文选用常见的两大评价指标,对聚类的结果进行评判。

1)NMI 评价指标[14]

$$\text{NMI}=\frac{\sum_{i=1}^{C}\sum_{j=1}^{C}N_{i,j}\log\frac{N\times N_{i,j}}{N_i\times N_j}}{\sqrt{(\sum_{i=1}^{C}N_i\log N_i/N)\times(\sum_{j=1}^{C}N_j\log N_j/N)}} \tag{14}$$

其中:$N_{i,j}$ 表示第 i 个聚类与类 j 的契合程度;N_i 表示第 i 个聚类所包含的数据样本量;N_j 表示类 j 所包含的数据样本量;N 表示整个数据样本的总量大小。

2)RandIndex 评价指标[14]

$$\text{RI}=\frac{f_{00}+f_{11}}{N(N-1)/2} \tag{15}$$

其中:f_{00} 表示数据点具有不同的类标签并且属于不同类的配对点数目;f_{11} 则表示数据点具有相同的类标签并且属于同一类的配对点数目;N 表示整个数据样本的总量大小。

3.3　模拟数据实验

本文采用高斯型函数生成相关的模拟数据集,模拟数据共分三大类,每类包含 200 个样本点,维数均为二维,此三类对应的类中心以及方差的取值如表 2 所示。在进行模拟实验时,max_iterative = 500,$\varepsilon=1e-7$,$\gamma=2$,$\eta=\frac{1}{N}$。

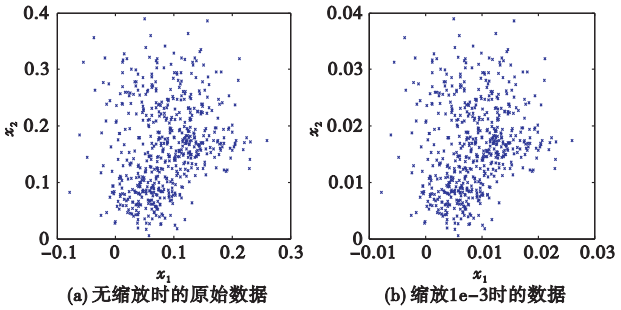
表 1　运行环境设置

运动环境	配置
硬件平台	CPU:Intel Pentium 双核 主频:1.6 GHz 内存:1 GB
编程环境	MATLAB 7.0

表 2　生成模拟数据集的各参数

类别	类中心	类方差
cluster 1	[5 8]	[10 0;0 10]
cluster 2	[11 16]	[25 0;0 7]
cluster 3	[9 25]	[30 0;0 20]

为了验证传统 MEC 算法在数据缩放时生产的中心一致性现象,根据结论 1 的分析,本文取缩放倍数 N 分别为 1、1e-1、1e-2 以及 1e-3 进行实验。图 2 给出相关的数据分布示意图。



(a)无缩放时的原始数据　　(b)缩放1e-3时的数据

图2　数据分布示意图

从图 2 可以明显地看出,图 2(a)与(b)除了数据上的等比例缩放外,形状以及所含的信息均无本质变化。从聚类角度看,这两个数据属于同源同信息的数据,因此利用聚类算法获取的类别划分也应该是一致的,且两者的聚类中心也应呈现等比例缩放的效果。为了求证以上论述,本文给出传统 MEC 算法以及本文算法在各缩放尺度下的聚类中心以及对应的聚类精度,如表 3、4 和图 3 所示。

分析以上结果可以发现,在数据无缩放时即($N=1$)的场景下,传统 MEC 算法可得到较好的聚类效果,然而在缩放尺度降至 0.1 以下时,MEC 算法已然失效,用该算法得到的聚类中心趋于一致。产生这种效果的原因正如 1.2 节的结论 1 所述,

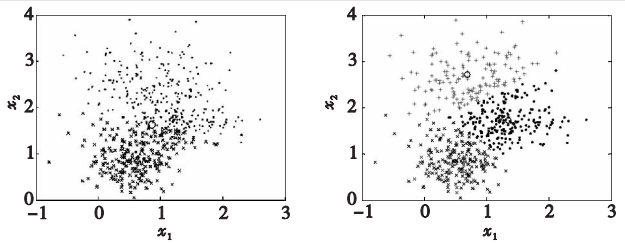
由于在缩放尺度降至 0.1 时,传统 MEC 算法中划分矩阵的求解受缩放尺度的影响非常大,两者之间呈现非线性关系,这就造成了再遇到这种特性的数据时,类别划分往往趋于一致,最终导致中心点一致,造成聚类精度直线下降,进而算法失效的后果。而本文算法(η-MCS-MEC)通过最大中心间隔项以及缩放因子的双重作用,成功地避免了中心一致现象的产生。通过观察表 3 可以发现,本文算法所得的类中心与缩放的数据呈线性增长关系,正是这种线性关系保证了本文算法不受数据变化的影响,而对此类变化数据具有鲁棒性。同时通过观察表 3 的第一列可以发现,本文算法在原始数据集上也比传统的 MEC 算法有着更好的聚类效果,其原因在于本文算法在原始算法的基础上成功地引入了中心最大间隔项,使得本文算法在处理类与类之间数据粘连的场景时,比起传统的算法有着更好的聚类效果,其受到粘连数据的影响较小,从而精度更高。

表 3 模拟数据集于各缩放尺度下各算法得到的聚类中心

algorithms	cluster	scaling			
		$N=1$	$N=1e-1$	$N=1e-2$	$N=1e-3$
MEC	cluster 1	[13.4866 17.2752]	[0.8640 1.6259]	[0.0864 0.1626]	[0.0086 0.0163]
	cluster 2	[6.9800 26.3814]	[0.8640 1.6259]	[0.0864 0.1626]	[0.0086 0.0163]
	cluster 3	[5.1381 8.7325]	[0.8640 1.6259]	[0.0864 0.1626]	[0.0086 0.0163]
η-MCS-MEC	cluster 1	[8.6638 14.4357]	[0.8664 1.4436]	[0.0866 0.1444]	[0.0087 0.0144]
	Cluster 2	[9.8659 22.3745]	[0.9866 2.2375]	[0.0987 0.2238]	[0.0099 0.0224]
	cluster 3	[4.9013 8.2451]	[0.4901 0.8245]	[0.0490 0.0825]	[0.0049 0.0082]

表 4 模拟数据集于各缩放尺度下各算法聚类精度

algorithms	type	scaling			
		$N=1$	$N=1e-1$	$N=1e-2$	$N=1e-3$
MEC	NMI	0.5536	0.4699	0.4213	0.0234
	RI	0.8060	0.7085	0.7062	0.3345
η-MCS-MEC	NMI	0.6335	0.6335	0.6335	0.6335
	RI	0.8552	0.8552	0.8552	0.8552



(a) MEC算法得到的聚类结果 (b)η-MCS-MEC算法得到的聚类结果

图3 各算法于缩放1e-1倍的数据集上聚类结果示意图

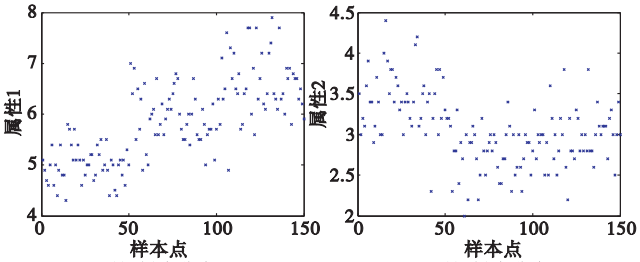
3.4 UCI 真实数据仿真实验

为进一步验证本文算法在真实数据集上的聚类效果,本文采用经典的数据源 UCI(University of California, Irvine)数据集[9]中的 IRIS 数据集进行仿真实验。该数据集的基本构成如表 5 所示。

表 5 真实数据集构成

name	cluster	dimension	size
Iris	3	4	150

由于 Iris 数据集的前两维样本的类间数据粘连也比较严重,如图 4 所示。正如模拟数据实验的结论,本文算法应比传统算法在该数据集上表现出更好的聚类效果。为了验证这一结论,本节将仿照模拟数据实验的架构对 Iris 数据集做各项针对性的仿真实验。具体在进行实验时,各参数的初始化设定值与模拟数据的实验设定方案一致。仿真实验结果如表 6 和 7 所示。



(a) 属性1样本分布 (b) 属性2样本分布

图4 Iris第1、2维的属性分布情况

表 6 不同缩放尺度下的各算法于 Iris 数据集上的聚类中心

algorithms	cluster	scaling			
		Iris	0.1 × Iris	0.01 × Iris	0.001 × Iris
MEC	cluster 1	[6.1267 2.9705 4.4676 1.4958]	[0.5843 0.3054 0.3759 0.1199]	[0.0584 0.0305 0.0376 0.0120]	[0.0058 0.0031 0.0038 0.0012]
	cluster 2	[5.2804 3.2200 2.3498 0.6082]	[0.5843 0.3054 0.3759 0.1199]	[0.0584 0.0305 0.0376 0.0120]	[0.0058 0.0031 0.0038 0.0012]
	cluster 3	[6.1263 2.9705 4.4668 1.4955]	[0.5843 0.3054 0.3759 0.1199]	[0.0584 0.0305 0.0376 0.0120]	[0.0058 0.0031 0.0038 0.0012]
η-MCS-MEC	cluster 1	[6.5053 2.9494 5.2427 1.8242]	[0.6505 0.2949 0.5243 0.1824]	[0.0650 0.0295 0.0524 0.0182]	[0.0065 0.0029 0.0052 0.0018]
	cluster 2	[5.0004 3.3882 1.4998 0.2600]	[0.5000 0.3388 0.1500 0.0260]	[0.0500 0.0339 0.0150 0.0026]	[0.0050 0.0034 0.0015 0.0003]
	cluster 3	[6.0746 2.8079 4.6621 1.5654]	[0.6075 0.2808 0.4662 0.1565]	[0.0608 0.0281 0.0466 0.0157]	[0.0061 0.0028 0.0047 0.0016]

表 7 不同缩放尺度下各算法于 Iris 数据集的聚类精度

algorithms	type	scaling			
		Iris	0.1 × Iris	0.01 × Iris	0.001 × Iris
MEC	NMI	0.6668	0.5979	0.5887	0.0817
	RI	0.8347	0.7433	0.7405	0.3482
η-MCS-MEC	NMI	0.7099	0.7099	0.7099	0.7099
	RI	0.8723	0.8723	0.8723	0.8723

根据表 6 和 7 的实验结果,进一步验证了模拟数据的结论是正确的。通过对 Iris 数据集的仿真实验也可得到如下的结论:a)在数据进行放缩时,传统 MEC 算法已然失效,特别是当缩放倍数从 N 降至 0.001 时,MEC 算法的聚类精度产生了骤降的现象;b)本文的方法再次得证,比传统的方法对数据的变化不敏感,并对该种变化具有鲁棒性,同时也进一步验证了本文的结论 1 是正确的,而本文方法亦行之有效地解决了类中心趋于一致的问题。

4 结束语

针对缩放型数据(或微量级数据)而言,传统的 MEC 算法已经无法适用,利用该算法得到的聚类中心往往趋于一致,最终导致算法的聚类效果急剧下降,通过研究分析可发现造成这类现象的原因在于传统 MEC 算法对变化的数据十分敏感,其变化规律与数据的缩放尺度呈非线性的关系。为了解决这一问题,本文提出了引入最大中心间隔项以及缩放因子 η 的新聚类算法 η-MCS-MEC。通过模拟数据以及 UCI 数据集的仿真实验结果均得到了与结论 1 一致的结果。同时又发现了本文算法对具有类间粘连的数据集比之传统 MEC 算法更加有效的结论。由于本文算法对数据所处的数量级并不敏感,对数据的变化也具有鲁棒性。因此该算法在微量级数据聚类或纳米级数据聚类分析时有很好的应用价值,利用该算法在处理上述数据时并不会随着数据量级的增加或下降而丢失精度。

(下转第 123 页)

表 4　两种算法 20 次实验迭代次数和收敛率的对比

函数	算法	收敛所需的最少迭代次数	收敛所需的最大迭代次数	收敛所需的平均迭代次数	收敛率
f_1	GSO	1	51	20.55	20/20
	GMGSO	1	20	7.45	20/20
f_2	GSO	457	605	531.84	19/20
	GMGSO	49	272	140.55	20/20
f_3	GSO	209	951	540.67	9/20
	GMGSO	46	286	193.45	20/20
f_4	GSO	491	608	535.10	20/20
	GMGSO	56	117	87.90	20/20
f_5	GSO	–	–	–	0/20
	GMGSO	42	87	61.85	20/20
f_6	GSO	–	–	–	0/20
	GMGSO	352	352	352	1/20

表 5　两种算法 20 次实验最优解的对比

函数	算法	最优值	最差值	平均值
f_1	GSO	1.14366661e－011	2.70051296e－007	3.91198785e－008
	GMGSO	2.78622804e－015	1.41708905e－009	2.45729840e－010
f_2	GSO	－0.51340894	－0.39474040	－0.50747581
	GMGSO	－0.51340925	－0.51340070	－0.51340734
f_3	GSO	－1.03162785	－0.215426809	－0.89907995
	GMGSO	－1.03162843	－1.03162793	－1.03162824
f_4	GSO	5.55736866e－019	6.60992810e－011	5.25539250e－012
	GMGSO	2.17635575e－078	3.46268213e－055	1.88164936e－056
f_5	GSO	0.16891981	0.85941366	0.47593719
	GMGSO	4.92552598e－009	3.18340614e－007	1.24322606e－007
f_6	GSO	10.15470648	29.54348965	17.84910058
	GMGSO	9.12832544e－006	0.03208201	0.00838718

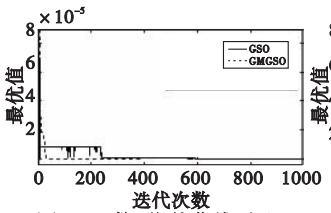


图1　函数f_1收敛曲线对比

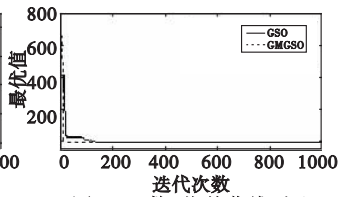


图2　函数f_2收敛曲线对比

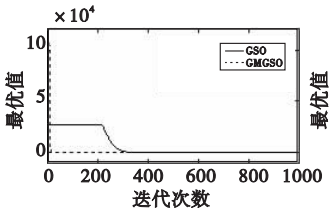


图3　函数f_3收敛曲线对比

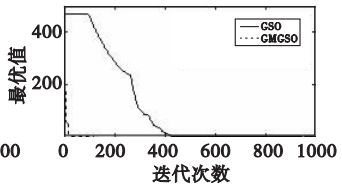


图4　函数f_4收敛曲线对比

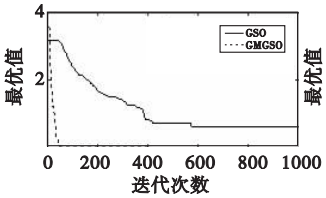


图5　函数f_5收敛曲线对比

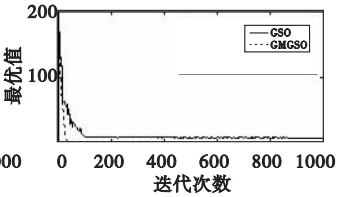


图6　函数f_6收敛曲线对比

4　结束语

本文提出了一种带高斯变异的人工萤火虫算法,通过在算法中加入高斯变异策略,有效防止了算法陷入局部最优值,提高了捕获到全局最优值的可能性,并提高了算法的收敛速度和求解精度。因此,改进后的 GMGSO 较 GSO 算法更有效,但改进后的算法对某些复杂函数的收敛率与求解精度仍有不足,还需要进一步研究。

参考文献:

[1] KRISHNANAND K N, GHOSE D. Glowworm swarm optimization: a new method for optimizing multi-modal functions [J]. International Journal of Computational Intelligence Studies, 2009, 1(1): 93-119.

[2] KRISHNANAND K N. Glowworm swarm optimization: a multimodal function optimization paradigm with applications to multiple signal source localization tasks [D]. Indian: Indian Institute of Science, 2007.

[3] KRISHNANAND K N, GHOSE D. A glowworm swarm optimization based multi-robot system for signal source localization [M]// LIU Di-kai, WANG Ling-feng, TAN K C. Design and Control of Intelligent Robotic Systems. Berlin: Springer, 2009:53-74.

[4] KRISHNANAND K N, GHOSE D. Chasing multiple mobile signal sources: a glowworm swarm optimization approach [C]//Proc of the 3rd Indian International Conference on Artificial Intelligence. [S. l.]: IEEE Press, 2007.

[5] 曲良东,何登旭. 基于自适应高斯变异的人工鱼群算法[J]. 计算机工程,2009,35(15):182-189.

[6] 黄凯,周永权. 带交尾行为的混沌人工萤火虫优化算法[J]. 计算机科学,2012,39(3):231-234.

[7] 黄凯,周永权. 一种改进的变步长自适应 GSO 算法[J]. 计算机工程,2012,38(4):185-187.

(上接第 106 页)

参考文献:

[1] HALL L O, GOLDGOF D B. Convergence of the single-pass and online fuzzy C-means algorithms[J]. IEEE Trans on Fuzzy Systems, 2011, 19(4):792-794.

[2] ZHU L, CHUNG F L, WANG S T. Generalized fuzzy C-means clustering algorithm with improved fuzzy partitions [J]. IEEE Trans on Systems Man and Cybernetics, 2009, 39(3):578-591.

[3] DENG Zhao-hong, CAI Ji-shi, CHUNG Fu-Lai, *et al*. Enhanced soft subspace clustering integrating within-cluster and between-cluster information[J]. Pattern Recognition, 2010, 43(3):767-781.

[4] RAJINI N H, BHAVANI R. Enhancing k-means and kernelized fuzzy C-means clustering with cluster center initialization in segmenting MRI brain images [C]//Proc of International Conference on Electronics Computer Technology. 2011:259-263.

[5] WU K L, YANG M S. Alternative C-means clustering algorithms[J]. Pattern Recognition, 2002, 35(10):2267-2278.

[6] YANG M S. On a class of fuzzy classification maximum likelihood procedures[J]. Fuzzy Sets and Systems, 1993, 57(3):365-375.

[7] LIN J S. Fuzzy clustering using a compensated fuzzy hopfield network [J]. Neural Processing Letters, 1999, 10(1):35-48.

[8] LI R P, MUKAIDON M. A maximum entropy approach to fuzzy clustering[C]//Proc of the 4th IEEE International Conference on Fuzzy System. 1995:2227-2232.

[9] KARAYIANNIS N B. MECA: maximum entropy clustering algorithm [C]//Proc of the 3rd IEEE Conference on Fuzzy Systems. 1994:630-635.

[10] WEI C, FAHN C. The multisynapse neural network and its application to fuzzy clustering[J]. IEEE Trans on Neural Networks, 2002, 13(3):600-618.

[11] 邓赵红,王士同,吴锡生,等. 鲁棒的极大熵聚类算法 RMEC 及其例外点标识[J]. 中国工程科学,2004,4(9):38-45.

[12] KRISHNAPURAM R, KELLER J M. A possibilistic approach to clustering[J]. IEEE Trans on Fuzzy Systems, 1993, 1(2):98-110.

[13] KRISHNAPURAM R, KELLER J M. The possibilistic means algorithms: insights and recommendation [J]. IEEE Trans on Fuzzy Systems, 1996, 4(3):98-110.

[14] LIU J, MOHAMMED J, CARTER J, *et al*. Distance-based clustering of CGH data[J]. Bioinformatics, 2006, 22(16):1971-1978.