

1-2-2019

Iterative Stochastic Resonance Demodulation Algorithm of Frequency-Hopping Signal

Haixia Li

1.National Key Laboratory of Electronic Measurement Technology, North University of China, Taiyuan 030051, China; ;2.Electronic Department, Xinzhou Teachers University, Xinzhou 034000, China;

Yongfeng Ren

1.National Key Laboratory of Electronic Measurement Technology, North University of China, Taiyuan 030051, China; ;

Yuhua Yang

1.National Key Laboratory of Electronic Measurement Technology, North University of China, Taiyuan 030051, China; ;

Zhang Baili

2.Electronic Department, Xinzhou Teachers University, Xinzhou 034000, China;

See next page for additional authors

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research](#), [Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Iterative Stochastic Resonance Demodulation Algorithm of Frequency-Hopping Signal

Abstract

Abstract: In order to improve the demodulation performance of frequency-hopping signals, this paper proposes a digital receiving algorithm based on the bistable stochastic resonance. *With the digital samples sifting, the algorithm processes multiple frequency signals in one stochastic resonance system. The conversion from channel noise to useful signal is improved with multiple iterations of signal system. Finally, the local signal is designed to eliminate the frequency distortion in stochastic resonance system through the relevant calculation.* Theoretical analysis and simulation results show that the algorithm can demodulate frequency-hopping signal, and its performance can be improved with the increase of sampling rate. With high sampling rate, the demodulation performance is 2 dB higher than the existing single random resonance detection system.

Keywords

wireless communication, stochastic resonance, iterative, frequency-hopping, sampling

Authors

Haixia Li, Yongfeng Ren, Yuhua Yang, Zhang Baili, and Zhumei Tian

Recommended Citation

Li Haixia, Ren Yongfeng, Yang Yuhua, Zhang Baili, Tian Zhumei. Iterative Stochastic Resonance Demodulation Algorithm of Frequency-Hopping Signal[J]. Journal of System Simulation, 2018, 30(1): 341-347.

一种基于变异蝙蝠算法的高维聚类方法

寇广^{1,2}, 汤光明¹, 何嘉婧¹, 张恒巍¹

(1. 中国人民解放军信息工程大学, 郑州 450001; 2. 信息保障技术重点实验室, 北京 100072)

摘要: 随着大数据时代的来临, 信息资源迅猛增长, 数据逐渐趋于高维化。传统的聚类方法针对低维数据有较好的效果, 而不再适用于高维数据。在目前已有的高维聚类算法的基础上, 提出一种基于智能优化算法的高维聚类算法 SSC-BA (Soft Subspace Clustering based on Bat Algorithm)。算法设计了一个新目标函数, 结合了加权类内相似性及类间差异性和界约束权值矩阵, 引进了一种变异蝙蝠算法计算权值矩阵, 给出了新的学习规则。对提出的算法进行了仿真实验, 与其他软子空间聚类算法进行对比测试。实验结果表明此聚类算法适用于高维数据并较其它算法有一定的性能优势。

关键词: 高维聚类; 权值矩阵; 蝙蝠算法; 变异策略

中图分类号: TP311

文献标识码: A

文章编号: 1004-731X (2018) 04-1253-07

DOI: 10.16182/j.issn1004731x.joss.201804006

High-dimensional Clustering Method Based on Variant Bat Algorithm

Kou Guang^{1,2}, Tang Guangming¹, He Jiajing¹, Zhang Hengwei¹

(1. The PLA Information Engineering University, Zhengzhou 450001, China;

2. Science and Technology on Information Assurance Laboratory, Beijing 100072, China)

Abstract: With the advent of the era of big data, the information resource is growing rapidly, and the data are becoming high-dimensional. Traditional clustering methods have a good effect for low-dimensional data, but no longer apply to high-dimensional data. On the basis of existing high-dimensional clustering algorithm, a high-dimensional clustering algorithm based on intelligent optimization SSC-BA is proposed. A novel objective function is designed, which integrates the fuzzy weighting within-cluster compactness and the between-cluster separation. A variant bat algorithm is introduced to calculate the weight matrix, giving the new learning rules. Simulation experiments are made for the proposed algorithm, and other soft subspace clustering algorithm is compared with the test. Experimental results show that the clustering algorithm is suitable for high-dimensional data, and has certain performance advantages compared with other algorithms.

Keywords: high-dimensional clustering; weighting matrix; bat algorithm; mutation strategy

引言

仿真科学在系统科学、控制科学、计算机科学

等学科中孕育发展, 已成功应用于各行各业。随着大数据时代的来临, 多维数据虚拟模型是还原物理世界的重要途径, 也是仿真科学的前沿课题。聚类仿真分析是此领域的重要方法。

聚类就是将物理或抽象的数据对象按照一定的相似性度量划分为不同类(簇)的过程, 是对数据进行处理的重要工具和方法。随着聚类技术的不断发展, 现在聚类分析在机器学习、数据挖掘等多个



收稿日期: 2016-04-14 修回日期: 2016-06-25;
基金项目: 国家自然科学基金(61303074), 信息保障技术重点实验室开放基金(KJ-14-106);
作者简介: 寇广(1983-), 男, 河南许昌, 博士, 副教授, 研究方向为网络安全态势感知、深度学习;
汤光明(1963-), 女, 博士, 教授, 研究方向为网络安全、信息隐藏。

<http://www.china-simulation.com>

• 1253 •

领域中发挥着越来越重要的作用^[1]。大部分聚类算法都是针对低维度的数据而设计,也能在低维数据上取得较好的聚类效果,然而很多算法受“维数灾难”的影响,在高维数据上的性能并不好^[2]。已有的高维算法可以分为特征转换和子空间聚类两种,特征转换指通过主成分分析或奇异值分解等策略,把原始高维数据线性合并至一个低维的空间,然后使用传统的聚类算法在该低维空间上进行聚类,以达到降维的目的。然而此种方法难以确定低维空间的维度而且在高维空间中大量噪音或冗余特征掩盖了簇,会给降维过程造成困难。子空间聚类只选取有效的特征组成相关的子空间,并在该子空间上执行聚类任务。它将数据样本划分成簇的同时,搜索各个簇所在的子空间^[3]。在每一个簇中,各个特征被赋予不同的权值,用于度量特征与簇的相关性。子空间聚类算法能有效减少冗余和不相关特征对聚类过程的干扰,从而提高高维数据集上的聚类效果。

根据加权方法的不同,子空间聚类可分为硬子空间聚类和软子空间聚类。在硬子空间聚类中,特征权值为0或1^[4];而在软子空间聚类中,给数据集中的各簇赋予不同的特征权重向量,用来表示聚类过程中各特征对此簇贡献的大小。相比硬子空间聚类,软子空间聚类反映了各个特征在与簇的相关性的差异,更能有效解决实际应用中的问题^[5]。软子空间聚类算法可分为基于局部搜索策略的软子空间聚类算法和基于全局搜索策略的软子空间聚类算法。其中基于局部搜索的几种代表性算法有模糊子空间聚类(Fuzzy Subspace Clustering, FSC)算法^[6]、熵加权k-均值(Entropy Weighting k-means, EWKM)算法^[7]和增强的软子空间聚类(Enhanced Soft Subspace Clustering, ESSC)算法^[8]等,基于全局搜索的代表性算法为Lu等^[9]通过引入CLPSO(Comprehensive Learning Particle Swarm Optimizer)算法提出的PSOVW(Particle Swarm Optimizer for Variable Weighting)算法,改变了聚类算法的搜索策略,提高了聚类效果。

蝙蝠算法(Bat Algorithm, BA)是剑桥大学学者Yang提出的一种模拟自然界蝙蝠通过超声波搜索原理进行捕食的启发式群智能优化算法^[10]。与粒子群算法类似,蝙蝠算法也是基于群体的全局随机搜索机制,并采用速度、位移模型,避免了复杂的遗传操作。区别在于蝙蝠算法的随机性更强,因此蝙蝠算法的收敛速度快,鲁棒性好,而且具有模型简单,易于实现,结构简洁、潜在并行性和分布式等特点,已被证明在收敛速度和稳定性方面均超过了粒子群优化算法。

本文提出一种高维聚类算法SSC-BA(Soft Subspace Clustering based on Bat Algorithm,基于蝙蝠算法的软子空间聚类算法)。首先对聚类的目标函数加以改进,增加了类间差异度对聚类结果的影响,将权值矩阵约束放松为界约束,然后引入一种变异的蝙蝠算法优化新目标函数和搜索子空间中的聚类。实验表明,新的目标函数和变异蝙蝠算法的引入使算法较其他软子空间聚类算法有优势。

1 变异蝙蝠算法

1.1 基本蝙蝠算法

该算法将搜索过程模拟成蝙蝠个体搜寻猎物 and 移动过程。将蝙蝠个体映射到搜索空间中,将优化问题的解作为搜索空间中的一个蝙蝠,根据每个蝙蝠个体所处的位置计算出求解问题的目标函数值,蝙蝠群体通过调整频率、响度、脉冲发射率,追随当前最优蝙蝠,从而在解空间中进行搜索。

搜索过程中,第*i*只蝙蝠的脉冲频率的计算公式为:

$$f_i = f_{\min} + (f_{\max} - f_{\min}) \cdot \beta \quad (1)$$

式中: f_{\min} 和 f_{\max} 分别是脉冲频率的最小值和最大值, β 为随机数,在 $[0, 1]$ 上服从均匀分布。

蝙蝠的速度更新公式为:

$$v_i^{t+1} = v_i^t + (x_i^t - x_b) \cdot f_i \quad (2)$$

式中: v_i^{t+1} 和 v_i^t 分别为第*i*只蝙蝠在*t+1*代和*t*代的飞行速度; x_i^t 表示第*i*只蝙蝠在*t*代的位置; x_b 表示当前群体中最优蝙蝠的位置。

蝙蝠的位置更新公式为:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3)$$

其中, x_i^{t+1} 和 x_i^t 分别为第 i 只蝙蝠在 $t+1$ 代和 t 代的位置。

蝙蝠的脉冲频度和脉冲音强更新公式分别为:

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma \cdot t)] \quad (4)$$

$$A_i^{t+1} = \alpha \cdot A_i^t \quad (5)$$

式中: r_i^{t+1} 表示第 i 只蝙蝠在 $t+1$ 代的脉冲频度特征; r_i^0 表示第 i 只蝙蝠的最大脉冲频度; $\gamma > 0$ 为脉冲频度增加系数; A_i^{t+1} 和 A_i^t 分别表示第 i 只蝙蝠在 $t+1$ 和 t 代发射脉冲的音强; $\alpha > 0$ 为脉冲音强衰减系数。

1.2 蝙蝠算法的变异

基本 BA 算法是一种性能良好的全局优化算法, 但是也存在后期易陷入局部最优的问题^[11]。本文对群体中当前全局最优解进行多样性变异操作^[12], 变异的好处在于避免所有蝙蝠个体均向全局最优解方向飞去, 导致群体多样性损失的现象, 同时避免算法收敛速度明显变慢或停止, 增加种群的多样性后, 算法不易陷入局部最优。变异后的蝙蝠算法能够更加准确地寻找到最优值, 从而取得更好的聚类效果。

假设当前全局最优解为 $\mathbf{x} = (x_1, x_2, \dots, x_n)$, 以概率 $1/n$ 随机从 \mathbf{x} 中选择一个元素 $x_k (k=1, 2, \dots, n)$, 然后在范围 $[l_i, h_i]$ 内随机产生一个实数替代 \mathbf{x} 中的元素, 从而产生一个新的个体 $\mathbf{x}' = (x'_1, x'_2, \dots, x'_n)$ 。

变异算子的计算公式为:

$$x'_i = \begin{cases} l_i + \lambda(h_i - l_i), & i = k \\ x_i, & \text{otherwise} \end{cases} \quad (6)$$

式中: λ 为随机数, 在 $[0, 1]$ 上服从均匀分布。

2 基于变异蝙蝠算法的软子空间聚类算法

2.1 目标函数

Lu 等^[9]指出, 合适的目标函数和有效的搜索策略是提高软子空间聚类算法性能的基础。本文将类间差异度综合在目标函数中, 从而得到一个结合

类内相似度、类间差异度与界约束权值矩阵的目标函数如式(7):

$$J_{SSC-BA} = \sum_{i=1}^Q \sum_{k=1}^D \sum_{j=1}^N u_{ij} \left(\frac{w_{ik}}{D} \right)^r (x_{jk} - c_{ik})^2 - \eta \sum_{i=1}^Q \left(\sum_{j=1}^N u_{ij} \right) \sum_{k=1}^D \left(\frac{w_{ik}}{D} \right)^r (c_{ik} - c_{0k})^2 \quad (7)$$

$$\text{s.t. } 0 \leq u_{ij} \leq 1, \sum_{i=1}^Q u_{ij} = 1, 0 < \sum_{j=1}^N u_{ij} < N, 0 \leq w_{ik} \leq 1$$

式中: $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_Q] (\mathbf{c}_i = [c_{i1}, \dots, c_{iD}])$ 是聚类中心矩阵; $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N] (\mathbf{u}_j = [u_{1j}, \dots, u_{Qj}])$ 是模糊划分矩阵, u_{ij} 代表个体 j 对簇 i 的隶属度; $\mathbf{W} = [\mathbf{w}_1, \dots, \mathbf{w}_Q] (\mathbf{w}_i = [w_{i1}, \dots, w_{iD}])$ 是权值矩阵; 参数 $\eta \in (0, 1)$ 用来平衡类内相似度和类间差异度, η 越大, 类间差异度占的比重越大, η 越小, 类内相似度占的比重越大; c_{0k} 代表初始聚类中心, 它的计算公式为:

$$c_{0k} = \frac{\sum_{j=1}^N x_{jk}}{N} \quad (8)$$

2.2 学习规则

聚类的过程就是将式(7)中的目标函数最小化的过程, 即最小化类内模糊加权距离并最大化类间加权距离。根据拉格朗日定理, 可以得到 SSC-BA 的聚类中心更新公式如式(9):

$$c_{ik} = \frac{\sum_{j=1}^N u_{ij} (x_{jk} - \eta c_{0k})}{\sum_{j=1}^N u_{ij} (1 - \eta)} \quad (9)$$

在高维稀疏数据集上, 样本点之间的距离非常小, 一个样本到各个簇的隶属度趋同, 簇的划分极不明显, 模糊聚类算法会寻找整个数据集的相关特征而不是寻找每一个簇的相关特征, 而且模糊聚类的运算时间随着维数的增加剧增。而硬聚类能敏感地捕捉到样本点之间的距离差异, 因此本文采用硬划分的方法计算划分矩阵, 其更新公式如式(10):

$$u_{ij} = \begin{cases} 1, i = \arg \min_{q=1, \dots, C} d_{jq} \\ 0, \text{else} \end{cases} \quad (10)$$

采用全局搜索的思想, 将 SSC-BA 中权值矩阵 \mathbf{W} 看作是算法的解, 运用变异蝙蝠算法进行更新。BA 是一种简单有效且全局优化能力较强的全局搜索算法, 一个蝙蝠位置 $\mathbf{x}_i^t = \{x_{i1}^t, \dots, x_{id}^t, \dots, x_{iC \times D}^t\}$ 对应于一个权值矩阵 \mathbf{W} , 其中每 D 维对应于 D 个特征在一个簇中的权值。同时, 算法为每一个位置保留其对应的聚类中心矩阵 \mathbf{C} 以及划分矩阵 \mathbf{U} 。

在算法的初始化阶段, 要选择 Q 个样本作为初始聚类中心。一般的软子空间聚类算法均采用随机选取的方式, 会造成算法对初始值条件很敏感, 为了减弱敏感性, 本文采用如下的方法:

首先为每一只蝙蝠随机选取 Q 个样本作为聚类中心, 选取 M 次, 得到 M 个聚类中心矩阵 \mathbf{V} , 令 $\mathbf{W} = [w_{ik}]_{C \times D} = [1/D]_{C \times D}$, 利用式(10)计算出划分矩阵 \mathbf{U} , 然后根据式(7)计算出目标函数并从小到大排序, 取值最小的 Q 个样本作为 Q 个初始聚类中心, 得到它们的划分矩阵和目标函数。如此初始聚类中心的选取减弱了随机性, 使算法结果受初始值选取影响不大。

然后根据式(1)、(2)和(3)更新每个蝙蝠的速度和位置。产生一个随机数 $\text{rand}1$, 如果 $\text{rand}1 > r_i$, 则对当前最优蝙蝠位置进行扰动得到新的位置, 然后替换原位置; 产生一个随机数 $\text{rand}2$, 如果 $\text{rand}2 > A_i$ 且 $f(x_i) < f(x_b)$, 则移动至更新后的位置。根据式(4)和(5)更新脉冲频率 r 和脉冲音强 A 。最后以一定概率根据式(6)对当前全局最优蝙蝠的位置进行变异操作。计算目标函数值, 找出当前最优蝙蝠后再次进行迭代, 直到满足终止条件。最终输出的全局最优值即为最终的权值矩阵。

2.3 算法流程

SSC-BA 算法输入种群规模 n 、最大迭代次数 MaxIter 、聚类数 Q 、样本维数 D 及各个参数的值, 将变异蝙蝠算法引入软子空间聚类算法之中, 在对权值矩阵 \mathbf{W} 用 BA 算法进行更新的同时, 对聚类

中心矩阵 \mathbf{C} 和划分矩阵 \mathbf{U} 同时进行更新, 最终迭代得到最终的 \mathbf{U} 、 \mathbf{C} 、 \mathbf{W} 即为最终的聚类结果。具体的流程如图 1 所示。

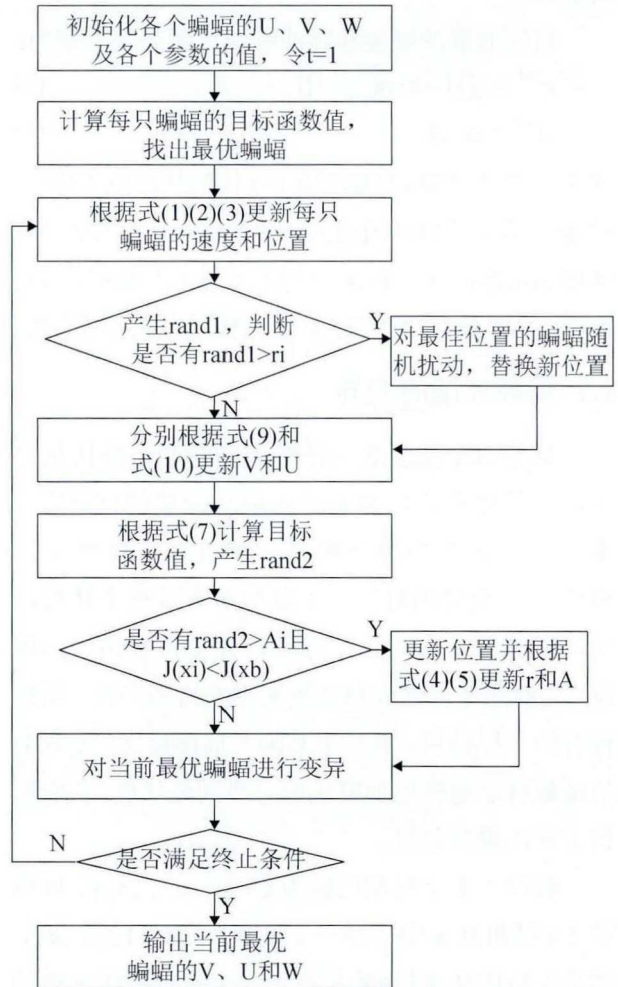


图 1 SSC-BA 算法的基本流程

Fig. 1 The basic flow of the SSC-BA algorithm

3 仿真实验与分析

3.1 实验设置

本文提出的 SSC-BA 算法采用经典测试集 UCI 和生物学中的癌基因表达数据集进行性能测试聚类分析。其中 UCI 测试集是用来测试 SSC-BA 对低维数据的兼容性, 癌基因表达数据集具有样本数量少但特征数目多的特点, 用来测试其高维数据中的适用性。测试的硬件环境为 2.66GHz CPU, 4 GB 内存, 所有算法在 WEKA 平台^[13]下开发。每

种算法分别使用不同的参数来实验, 聚类过程重复 20 次, 计算出聚类结果的平均值和标准差, 选择参数设置的最好结果参与比较。选取的数据集详细信息见表 1。

表 1 数据集详细信息
Tab. 1 Data set details

Dataset	Size of dataset	Number of dimensions	Number of clusters
Iris	150	4	3
Vernicle	846	18	4
Glass	214	9	6
CNS	34	7 129	2
Leukemia-MLL	72	12 582	3
Prostate	33	12 626	2

为了验证 SSC-BA 的算法性能, 对其和其它四个软子空间聚类算法 FSC、EWKM、ESSC、PSOVM 进行比较。实验中的每个特征的值都标准化为 [0,1] 区间, 由于 SSC-BA 和 PSOVM 是基于群体的算法, 而 FSC 等是单点搜索算法, 本文采用 MaxIter 作为所有测试算法的终止条件。令 MaxIter 为 100, 所有基于群体算法的种群规模为 20。其余参数设置如表 2 所示。

表 2 参数设置
Tab. 2 Preferences

Algorithm	Parameters
SSC-BA	$r_0=0.75, A_0=0.25, \gamma=0.05, \alpha=0.95$ $\tau=2, \eta=0.1, 0.2, 0.3, 0.5, 0.9$
FSC	$\epsilon_0=0, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-5}, \tau=1.2, 1.5, 2.5, 10$
EWKM	$\gamma=0.5, 1, 2, 5, 10$
ESSC	$m=(\min(N, D-1))/(\min(N, D-1)-2)$ $\gamma=0.5, 1, 2, 5, 10, \eta=0.1, 0.2, 0.3, 0.5, 0.9$
PSOVM	$\tau=3, 4, 5, 6, 7, 8$

本文采用两种性能指标对聚类效果进行评价: RI (Rand Index) 和 NMI (Normalized Mutual Information)^[14], 其中 RI 定义如下:

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

式中: f_{00} 是不属于同一个类并被分配到不同簇的样本点对的数量, f_{11} 是属于同一个类并被分配到同

一个簇的样本点对的数量。NMI 的定义如下:

$$NMI = \frac{\sum_{i=1}^K \sum_{j=1}^C n_{ij} \log \frac{N \cdot n_{ij}}{n_i \cdot n_j}}{\sqrt{\left(\sum_{i=1}^K n_i \log \frac{n_i}{N} \right) \left(\sum_{j=1}^C n_j \log \frac{n_j}{N} \right)}} \quad (12)$$

式中: K 是类别数; C 是聚类数, 依据数据类别数来确定聚类数, $K=C$; N 是样本总数; n_i 是属于类 i 的样本数; n_j 是属于簇 j 的样本数; n_{ij} 是属于类 i 并被分配到簇 j 的样本数。两个性能指标越大越好^[14]。

3.2 结果分析

表 3~4 中, 前三个数据为低维数据, 后三个数据为高维数据。Mean 和 Std 分别代表 20 次运算的平均值和标准差, 每一个数据集的最好结果在表中加粗表示, 比较直观地反映了各算法的聚类效果: SSC-BA 的 RI 和 NMI 值几乎在几个算法中最大, 标准差最小; ESSC 的 RI 和 NMI 值结果仅次于 SSC-BA, 但是标准差稍大; PSOVM 的 RI 和 NMI 值略优于 EWKM, 标准差接近于 SSC-BA; FSC 的 RI 和 NMI 值最小, 而且标准差也大于其他算法。综上可知, SSC-BA 算法在低维数据聚类中能取得很好的效果, 且优于其他的算法; 在高维数据中, 无论是 RI 还是 NMI, SSC-BA 基本是所有算法中效果最好的算法。无论是搜索策略、还是新的目标函数, 都有效地提高了软子空间聚类算法的效果。

与 FSC、EWKM 等 KM 型软子空间聚类算法相比, SSC-BA 基于模糊聚类对其进行了扩展。模糊聚类能很好地反映实践中不同簇之间往往并没有明确的分界这一现实, 因而往往能获得比硬聚类更好的效果, 并拥有不依赖于初始解以及不易陷入局部最优的优点; 与 ESSC 算法相比, SSC-BA 将 W 的等式约束放松为界约束, 并且采用了全局搜索策略, 使 W 的更新不受 U 和 C 的影响; 与 PSOVM 相比, SSC-BA 在聚类过程中考虑了类间差异性, 聚类的结果更加准确, 而且 BA 算法结构简单, 控制参数少, 性能优于标准 PSO, 有更强的鲁棒性和遗传性, 而且 SSC-BA 更易实现, 更具有高效性。

表 3 五种聚类算法 20 次运行的实验结果(RI)
Tab. 3 The experimental results of the 20 operation from five clustering algorithms(RI)

Dataset		SSC-BA	FSC	EWKM	ESSC	PSOVM
Iris	Mean	0.935 7	0.836 2	0.876 3	0.873 4	0.854 3
	Std	0.002 3	0.071 1	0.003 3	0.004 1	0.006 4
Vernicle	Mean	0.643 7	0.657 2	0.650 1	0.655 8	0.592 1
	Std	0.001 8	0.028 7	0.004 7	0.008 6	0.004 3
Glass	Mean	0.700 7	0.653 4	0.661 0	0.699 0	0.612 9
	Std	0.019 9	0.016 6	0.076 0	0.028 4	0.027 2
CNS	Mean	0.575 1	0.572 1	0.565 5	0.581 3	0.522 4
	Std	0.024 3	0.035 7	0.041 0	0.038 4	0.039 7
Leukemia-MLL	Mean	0.778 3	0.751 2	0.749 1	0.760 2	0.766 4
	Std	0.005 8	0.024 5	0.004 7	0.011 5	0.003 2
Prostate	Mean	0.828 3	0.610 2	0.724 1	0.812 4	0.813 3
	Std	0.103 4	0.113 8	0.111 2	0.123 7	0.082 7

表 4 五种聚类算法 20 次运行的实验结果(NMI)
Tab. 4 The experimental results of the 20 operation from five clustering algorithms(NMI)

Dataset		SSC-BA	FSC	EWKM	ESSC	PSOVM
Iris	Mean	0.833 3	0.695 7	0.741 3	0.742 4	0.726 8
	Std	0.004 9	0.102 7	0.006 1	0.005 4	0.007 3
Vernicle	Mean	0.150 1	0.149 7	0.119 6	0.148 3	0.125 4
	Std	0.031 5	0.057 9	0.034 0	0.037 2	0.032 4
Glass	Mean	0.359 7	0.240 3	0.344 6	0.350 6	0.348 3
	Std	0.031 4	0.113 5	0.023 4	0.036 7	0.033 5
CNS	Mean	0.147 2	0.012 4	0.125 2	0.135 9	0.072 3
	Std	0.031 8	0.009 1	0.058 1	0.085 7	0.035 8
Leukemia-MLL	Mean	0.526 1	0.459 2	0.509 3	0.534 5	0.527 9
	Std	0.024 8	0.052 6	0.083 3	0.044 2	0.015 7
Prostate	Mean	0.683 7	0.227 8	0.472 7	0.657 9	0.667 3
	Std	0.052 8	0.077 6	0.069 1	0.140 7	0.056 1

4 结论

仿真科学是认识和改造客观世界的最经济、最稳妥的方法。线性和非线性方程组合、参数拟合、最优化等各种数值求解算法是仿真科学的重要研究内容。大数据时代,为了解决传统聚类方法不能解决高维数据聚类的问题,本文在已有的高维聚类算法的基础上提出了 SSC-BA 算法,是一种软子空间聚类算法。算法在加权目标函数的基础上结合加权类内相似性及类间差异性和界约束权值矩阵提出了一个新的目标函数,并对基本蝙蝠算法进行变异,引入目标函数并搜索子空间中的聚类。算法的

创新性在于高维聚类目标函数的设计及引入变异的 BA 进行特征权值寻优提高聚类性能。将 SSC-BA 算法与其他软子空间聚类算法进行对比测试,实验结果表明,该算法较其它几种算法聚类的性能和效果有了明显提高。

参考文献:

- [1] 李鑫,张继福,蔡江辉.一种基于大密度区域的模糊聚类算法[J].小型微型计算机系统,2012,33(6):1310-1315.
Li Xin, Zhang Jifu, Cai Jianghui. A Fuzzy Clustering Algorithm Based on Large Density Region[J]. Journal of Chinese Computer Systems, 2012, 33(6): 1310-1315.
- [2] WANG Yunyun, CHEN Songcan, XUE Hui, et al.

<http://www.china-simulation.com>

• 1258 •

- Semisupervised classification learning by discrimination-aware manifold regularization[J]. *Neurocomputing* (S0925-2312), 2015, 147(7): 299-306.
- [3] 陈黎飞, 郭躬德, 姜青山. 自适应的软子空间聚类算法[J]. *软件学报*, 2010, 21(10): 2513-2523.
LF Chen, GD Guo, QS Jiang. Adaptive Algorithm for Soft Subspace Clustering[J]. *Journal of Software*, 2010, 21(10): 2513-2523.
- [4] Yip K Y, Cheung D W, Ng M K: A practical projected clustering algorithm[J]. *IEEE Trans. Knowl. Data Eng* (S0169-023X), 2004, 16(11): 1387-1397.
- [5] Liang, B, et al. A novel attribute weighting algorithm for clustering high-dimensional categorical data[J]. *Pattern Recognition*(S0031-3203), 2011, 44(12): 2843-2861.
- [6] 徐小龙, 李永萍. 一种基于 MapReduce 的知识聚类与统计机制[J]. *电子与信息学报*, 2016, 38(1): 202-208.
XU Xiaolong, LI Yongping. Knowledge Clustering and Statistics Based on MapReduce. *JEIT*, 2016, 38(1): 202-208.
- [7] Jing L P, Ng M K, Huang Z X. An Entropy Weighting K-means algorithm for subspace clustering of high-dimensional sparse data[J]. *IEEE Transactions on Knowledge Data and Engineering*(S1041-4347), 2007, 19 (8): 1026-1041.
- [8] Deng Zhao-hong, Kup-Sze Choi, Chung Fu-Lai et al. Enhanced soft subspace clustering integrating within-cluster and between-cluster information[J]. *Pattern Recognition*(S0031-3203), 2010, 43(3): 767-781.
- [9] Lu Y, Wang S, Li S, et al. Particle swarm optimizer for variable weighting in clustering high-dimensional data[J]. *Machine Learning*(S0885-6125), 2011, 82(1): 43-70.
- [10] Yang X S. *Nature inspired meta-heuristic algorithms*[M]. 2nd ed. Frome, UK: Luniver Press, 2010: 97-104.
- [11] 尹进田, 刘云连, 刘丽, 等. 一种高效的混合蝙蝠算法[J]. *计算机工程与应用*, 2014, 50(7): 62-66.
YIN Jintian, LIU Yunlian, LIU Li, et al. Efficient hybrid bat algorithm[J]. *CEA*, 2014, 50(7): 62-66.
- [12] NAJMEH SADAT JADDI, SALWANI ABDULLAH, ABDUL RAZAK HAMDAN. Multi-population cooperative bat algorithm-based optimization of artificial neural network model[J]. *Information Sciences* (S0020-0255), 2015, 294(2): 628-644.
- [13] Hall M, Frank E, Holmes G, et al. The WEKA data mining software: an update[J]. *Sigkdd Explorations* (S1931-0153), 2009, 11(1): 10-18.
- [14] Liu J, Mohammed J, Carter J, et al. Distance-based clustering of CGH data[J]. *Bioinformatics* (S1367-4803), 2006, 22 (16): 1971-1978.

———
(上接第 1252 页)

- [16] Henein C M, White T. Macroscopic Effects of Microscopic Forces Between Agents in Crowd Models [J]. *Physica A* (S0378-4371), 2007, 373: 694-712.
- [17] 苗志宏, 李智慧. 火灾环境下人员疏散耦合 SPH 模型及仿真[J]. *系统仿真学报*, 2016, 28(2): 292-300.
Miao Z H, Li Z H. Coupling SPH Model for Occupant Evacuation in Fires and Simulation[J]. *Journal of System Simulation*, 2016, 28(2): 292-300.
- [18] Guo R Y. Simulation of Spatial and Temporal Separation of Pedestrian Counter Flow Through a Bottleneck [J]. *Physica A* (S0378-4371), 2014, 415: 428-439.
- [19] 何流, 李旭宏, 卢静. 基于动态广义势能场的行人微观模型及仿真[J]. *系统仿真学报*, 2014, 26(11): 2558-2563.
He L, Li X H, Lu J. Microscopic Pedestrian Modeling and Simulation Based on Dynamic Generalized Potential Field [J]. *Journal of System Simulation*, 2014, 26(11): 2558-2563.
- [20] 赵巍, 刘畅, 廉兴宇, 等. 人群运动仿真和疏散优化方法设计与实现[J]. *系统仿真学报*, 2014, 26(3): 523-529.
Zhao W, Liu C, Lian X Y, et al. Simulation of Crowd Movement and Design and Implementation of Evacuation Optimization Method [J]. *Journal of System Simulation*, 2014, 26(3): 523-529.
- [21] van den Berg J, Lin M, Manocha D. Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation [J]. *IEEE International Conference on Robotics and Automation* (S1050-4729), 2008: 1928-1935.