

# 基于改进 BA 算法的 K-means 聚类

杨菊靖, 张达敏

(贵州大学 大数据与信息工程学院, 贵阳 550025)

**摘要:** 针对传统 BA(蝙蝠)算法易被局部极值吸引、发生过早收敛等问题,将莱维飞行搜索策略引入传统 BA 算法对蝙蝠的位置和速度更新方式进行改进,从而提高算法的全局搜索能力;通过引入非线性惯性权重平衡算法的全局和局部搜索能力并提高算法搜索精度;结合 limit 阈值的思想避免算法过快陷入局部极值。通过对六个标准测试函数的实验表明,改进后的 BA 算法不仅在全局搜索能力上有所提高,而且具有较好的搜索精度。最后将改进后的 BA 算法同 K-means 聚类算法进行结合,提出了一种基于改进 BA 算法的 K-means 聚类算法。实验结果表明,改进的算法提高了聚类准确率及算法鲁棒性。

**关键词:** 蝙蝠算法; 莱维飞行; 惯性权重; limit 阈值; K-means 算法

**中图分类号:** TP393.04      **文献标志码:** A      **文章编号:** 1001-3695(2018)05-1454-04

**doi:**10.3969/j.issn.1001-3695.2018.05.038

## K-means clustering algorithm based on improved BA algorithm

Yang Juqing, Zhang Damin

(College of Big Data & Information Engineering, Guizhou University, Guiyang 550025, China)

**Abstract:** The traditional bat algorithm is easy to fall into local optimum and premature convergence solution, in order to improve the global search capability of the bat algorithm, this paper considered to take Lévy flight search strategy into the bat algorithm to update bat's position and velocity, at the same time by introducing the nonlinear inertia weight to balance the global and local search capability to improve the search precision of the algorithm. Then it combined the limit threshold theory to avoid getting trapped into local optima. The results on six standard test functions show that the improved bat algorithm not only improves the global search ability and has better accuracy. Finally, combining the improve bat algorithm with K-means clustering algorithm, this paper proposed a K-means clustering algorithm based on the improved bat algorithm. The experimental results show that the algorithm improves the clustering quality and the robustness of the algorithm.

**Key words:** BA algorithm; Lévy flight; inertia weight; limit threshold; K-means algorithm

聚类分析是一种重要的数据挖掘技术,其目的是将数据集合成若干类,使得同一个类内样本间的相似度尽可能大,而不同类间样本相似度尽可能小<sup>[1]</sup>。目前聚类分析已经广泛地应用于许多应用领域,包括商务智能、图像模式识别、Web 搜索、生物学等<sup>[2]</sup>。现今越来越多的学者将人工智能应用到聚类问题中,将群体智能优化算法同聚类算法进行结合,弥补了传统聚类算法的缺陷<sup>[3]</sup>。蝙蝠算法(BA)是 Yang 教授<sup>[4]</sup>于 2010 年提出的一种通过模拟蝙蝠回声定位行为来搜索全局最优解的新型群体智能优化算法。蝙蝠算法也是一种群体智能优化算法,具有较强的随机搜索能力,且具有收敛速度快、鲁棒性好的优点<sup>[5]</sup>。由于蝙蝠算法具有概念简单、易于实现等优势,使得该算法在多个学科和工程领域得到了广泛应用<sup>[6,7]</sup>。

本文针对传统蝙蝠算法易被局部极值吸引、发生过早收敛等问题,将 Lévy 飞行搜索模式引入蝙蝠算法,通过利用 Lévy 飞行能够产生较大跳跃的这种不均匀随机游走的特性,提升算法的全局寻优能力和搜索精度;受 PSO 算法的启发,对 BA 算法引入非线性惯性权重因子来达到平衡全局搜索和局部搜索目的,防止过早收敛;为了提高算法跳出局部最优的能力,引入 limit 阈值思想,避免算法过早陷入局部最优。最后将改进后的蝙蝠算法同 K-means 聚类算法进行结合,提出一种基于改进 BA

算法的 K-means 聚类,实现聚类结果的优化,提高聚类质量。

## 1 蝙蝠算法

### 1.1 经典 BA 算法

蝙蝠算法是通过模拟现实生活中蝙蝠的回声定位搜索行为的一种群体智能优化算法。蝙蝠在搜索过程中本质上是通过对频率  $f$  对蝙蝠的移动步伐和范围的控制来进行速度和位置更新的;蝙蝠在寻优的过程中,通过对脉冲发射率  $r$  和声音响度  $A$  的调节寻找最优解,开始时,蝙蝠只具有较小的脉冲发射率和较大的响度,随着迭代的增加,脉冲发射率增加,响度减小,蝙蝠不断对目标进行扫描定位,最终搜索得到最优解。

经典蝙蝠算法流程如图 1 所示。

蝙蝠算法中对蝙蝠位置和速度的更新采取以下公式:

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

$$v_i^t = v_i^{t-1} + (x_i^t - x^*) \times f_i \quad (2)$$

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

其中:  $\beta \in [0, 1]$  是一个随机向量;  $v_i^t$  和  $v_i^{t-1}$  是蝙蝠在  $t$  代和  $t-1$  代搜索过程中的速度;  $x_i^t$  和  $x_i^{t-1}$  是蝙蝠在  $t$  代和  $t-1$  代搜索

收稿日期: 2017-01-09; 修回日期: 2017-02-21

**作者简介:** 杨菊靖(1993-),女,贵州贵阳人,硕士研究生,主要研究方向为数据挖掘(1029735033@qq.com);张达敏(1967-),男,贵州贵阳人,教授,主要研究方向为计算机应用技术、网络拥塞控制、病毒传播机制。

过程中的位置; $x^*$ 是当前搜索过程中的全局最佳位置;频率 $f_i$ 的取值可根据所求解的实际问题进行确定。开始时每只蝙蝠的频率是随机分配的,频率为 $[f_{\min}, f_{\max}]$ 。在进行局部搜索时,每只蝙蝠的位置更新公式如下:

$$x_{\text{new}} = x_{\text{old}} + \varepsilon A^t \quad (4)$$

其中: $\varepsilon \in [-1, 1]$ 是随机数; $A^t$ 是所有蝙蝠在这一代里的平均响度。

同时,随着迭代过程的进行脉冲发射频率和响度也要进行更新,当蝙蝠发现猎物,蝙蝠发射的脉冲响度降低,频率升高。脉冲发射的响度 $A_i$ 和频率 $r_i$ 的更新公式如下:

$$A_i' = \alpha A_i^{t-1}, r_i' = r_i^0 [1 - e^{-\gamma}] \quad (5)$$

其中: $A_i'$ 和 $A_i^{t-1}$ 分别表示第 $i$ 只蝙蝠在 $t$ 和 $t-1$ 代搜索过程中的脉冲发射响度; $\alpha \in [0, 1]$ 表示脉冲发射响度衰减系数; $r_i'$ 表示第 $i$ 只蝙蝠在 $t$ 代搜索过程中的脉冲发射频率; $r_i^0$ 表示第 $i$ 只蝙蝠的最大脉冲发射频率; $\gamma$ 表示脉冲发射频率增加系数。

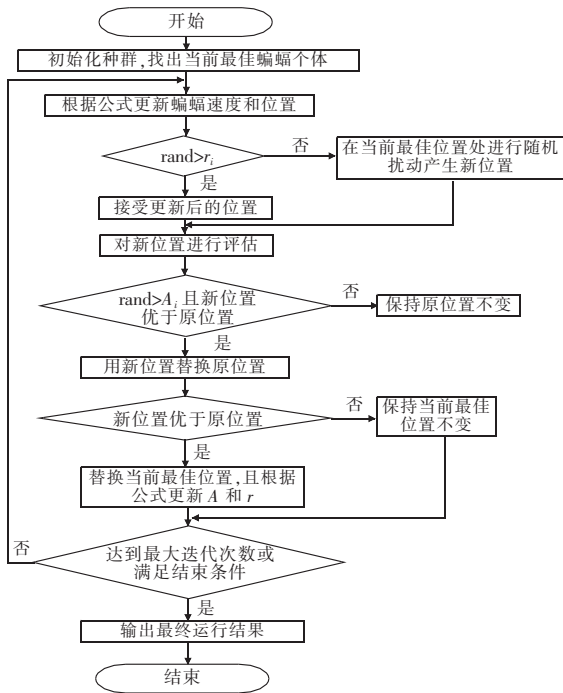


图1 经典蝙蝠算法流程

## 1.2 改进的 BA 算法

1) Lévy 飞行 Lévy 飞行的数学模型是由法国数学家 Lévy 提出, Benoist-Madelbrot 进行描述的一种稳定分布<sup>[8]</sup>。它是一种随机游走过程,行走的步长满足重尾分布<sup>[8]</sup>。研究表明, Lévy 飞行现象在自然界中十分普遍,许多动物或昆虫的行为都具有莱维飞行的特征,如蚂蚁、果蝇等。Lévy 飞行在搜索过程中会产生较大的跳跃,并且所运动的方向会产生急剧变化,这样可以使得算法跳出局部最优<sup>[9-11]</sup>,特别是在高维的复杂空间中,通过 Lévy 飞行策略可以扩大搜索范围,能够有效地避免收敛早熟,并且能提高算法的收敛速度<sup>[12]</sup>。

本文将 Lévy 飞行搜索策略引入蝙蝠算法,模拟蝙蝠的飞行搜索行为,对蝙蝠的位置和速度更新模式进行改进,利用 Lévy 飞行的特性来拓展搜索空间,从而提升蝙蝠算法的全局搜索能力及收敛精度。使用 Lévy 飞行策略对蝙蝠的位置和速度更新公式进行改进:

$$x_i' = x_i^{t-1} + (x_i^{t-1} - x^*) \otimes \text{Levy}(\lambda) \quad (6)$$

其中: $x_i'$ 表示蝙蝠 $i$ 在 $t$ 次搜索中的空间位置; $x_i^{t-1}$ 表示蝙蝠 $i$

在 $t-1$ 次搜索中的空间位置; $x^*$ 是当前搜索过程中的全局最佳位置;Levy( $\lambda$ )是步长服从 Lévy 分布的随机搜索向量; $\lambda$ 的取值为 $1 \leq \lambda \leq 3$ ;  $\otimes$ 表示矢量运算。

2) 非线性惯性权重因子 Shi 等人<sup>[13]</sup>在粒子群算法的速度公式中引入惯性权重,可达到提高收敛速度的效果。受粒子群算法的启发,由于蝙蝠算法是通过脉冲发射率 $r$ 控制算法的全局和局部搜索能力,在算法执行局部搜索时,随着迭代的进行脉冲发射率增加,算法逐渐失去局部搜索能力,导致算法收敛精度降低<sup>[14,15]</sup>。

为了更好地模拟现实生活中蝙蝠的搜索模式,经测试,发现采用非线性惯性权重比常数惯性权重的效果好,通过采用非线性惯性权重因子 $w$ 控制蝙蝠飞行位置变化幅度,从而在提高算法全局搜索能力的同时达到平衡算法的局部和全局搜索的目的。将速度和位置更新公式改为

$$x_i' = \omega_i \times x_i^{t-1} + (x_i^{t-1} - x^*) \otimes \text{Levy}(\lambda) \quad (7)$$

$$\omega_i = \omega_{\min} + (1 - \exp(\log(\text{iter}/\text{maxCycle}))) \times (\omega_{\max} - \omega_{\min}) \quad (8)$$

其中: $\omega_{\min}$ 表示最小惯性因子; $\omega_{\max}$ 表示最大惯性因子;iter 表示当前迭代次数;maxCycle 表示算法最大迭代次数。

3) limit 阈值 针对算法容易陷入局部最优,为了寻求全局最优解的问题,研究者常通过引入 limit 阈值对算法进行改进以达到克服局部最优,如人工蜂群算法在侦察蜂阶段引入了观察可行解的停滞次数 limit 阈值从而跳出局部最优<sup>[16,17]</sup>。在此,将阈值 limit 引入到蝙蝠算法中,在蝙蝠个体进行局部搜索时不仅可以通过脉冲发射率 $r$ 判断,还可以通过利用 limit 阈值进行判定,通过阈值 limit 限定算法产生随机扰动跳出局部极值,使得算法更好地跳出局部最优解。然而,limit 的值会影响算法的优化结果,limit 的值太小会导致随机更新操作太过于频繁,从而导致优化结果不好;反之,limit 的值太大则容易导致算法陷入局部最优。通过不断测试,在本文中将 limit 阈值设置为 10 较为理想。

## 1.3 实验仿真

为了验证改进后的 BA 算法性能,本文对六个标准测试函数进行实验仿真测试。测试时实验参数的设置如表 1 所示。实验中所采用的测试函数如表 2 所示。表 3 记录了实验仿真结果对比数据。图 2~7 显示了仿真实验中函数的适应度值随着迭代次数变化的曲线。

表1 实验参数设置

算法	$N$	$f_{\min}$	$f_{\max}$	$A$	$r$	$\alpha$	$\gamma$	maxCycle
BA	20	0	2	0.5	0.5	0.95	0.05	100
LWBA	20	0	2	0.5	0.5	0.95	0.05	100

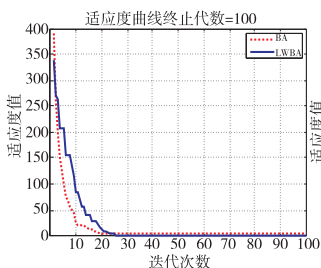
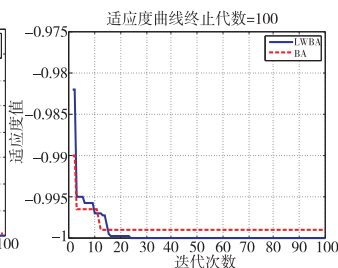
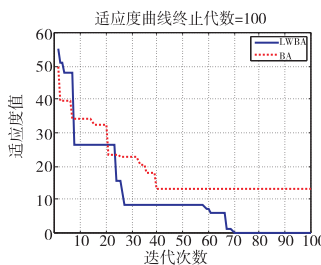
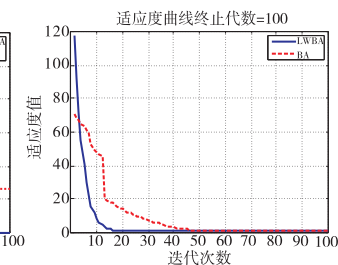
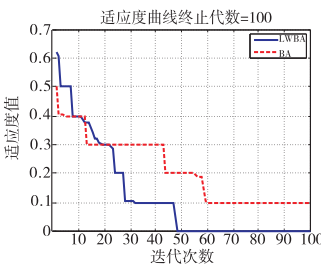
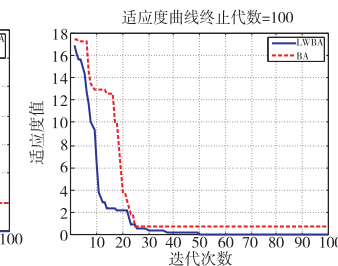
表2 标准测试函数

函数名	函数表达式	搜索空间	理论优解
sphere	$f_1(x) = \sum_{i=1}^n x_i^2$	$[-10, 10]$	$f_1(0, \dots, 0) = 0$
schaffer F6	$f_2(x) = \frac{\sin^2 \sqrt{x_1^2 + x_2^2} - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2} - 0.5$	$[-100, 100]$	$f_2(0, 0) = -1$
rastrigin	$f_3(x) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)]$	$[-5.12, 5.12]$	$f_3(0, \dots, 0) = 0$
griewank	$f_4(x) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	$[-600, 600]$	$f_4(0, \dots, 0) = 0$
salomon	$f_5(x) = -\cos(2\pi \sqrt{\sum_{i=1}^n x_i^2}) + 0.1 \sqrt{\sum_{i=1}^n x_i^2} + 1$	$[-5, 5]$	$f_5(0, \dots, 0) = 0$
ackley	$f_6(x) = -20 \exp[-\frac{1}{5} \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}] - \exp[\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)] + 20 + e$	$[-30, 30]$	$f_6(0, \dots, 0) = 0$

所采用的标准测试函数中,除函数 $f_1$ 是单峰函数外,其余 $f_2 \sim f_6$ 都是非线性的多峰多极值函数;函数 sphere 只有一个最小值;函数 schaffer F6 在搜索空间中具有强烈振荡的特点;函数 rastrigrin 是一种典型的非线性多模态函数,峰行呈现高低起伏不定跳跃,在解空间内大约有  $10n$  个局部极小点;函数 griewank 是典型的非线性多模态函数,具有广泛的搜索空间,它存在许多局部极小点,数目与问题的维数有关;函数 salomon 在以全局最小值为圆心、径向环形区域内存在无限多个局部极小点,函数振荡强烈;函数 ackley 是指数函数叠加上适度放大的余弦而得到的连续型实验函数,由于其具有一个几乎平坦的区域由余弦波调制形成一个孔或峰使得曲面起伏不平的特性,导致函数的搜索十分复杂。实验对每种算法独立运行 10 次,记录其运行结果如表 3 所示。

表3 算法 BA 及 LWBA 对函数的寻优结果对比

函数	维数	寻优结果		
		最优寻优值	平均寻优值	最高寻优值
$f_1(x)$	$D=20$	BA: 1.9153	BA: 2.27671	BA: 2.7138
		LWBA: 5.6066e-09	LWBA: 8.650301e-08	LWBA: 2.201e-07
$f_2(x)$	$D=2$	BA: -1	BA: -0.979316	BA: -0.89785
		LWBA: -1	LWBA: -1	LWBA: -1
$f_3(x)$	$D=5$	BA: 4.9698	BA: 12.55939	BA: 25.5594
		LWBA: 3.9088e-07	LWBA: 2.036352e-05	LWBA: 5.0577e-05
$f_4(x)$	$D=10$	BA: 1.1216	BA: 3.04527	BA: 6.0612
		LWBA: 5.0663e-05	LWBA: 0.46056	LWBA: 0.92796
$f_5(x)$	$D=5$	BA: 0.099875	BA: 0.02085832	BA: 0.0012903
		LWBA: 5.5008e-07	LWBA: 3.98819e-06	LWBA: 1.377e-05
$f_6(x)$	$D=5$	BA: 0.44246	BA: 0.614918	BA: 1.0493
		LWBA: 5.2573e-07	LWBA: 1.177453e-05	LWBA: 5.4509e-05

图2 函数 $f_1$ 寻优曲线对比图3 函数 $f_2$ 寻优曲线对比图4 函数 $f_3$ 寻优曲线对比图5 函数 $f_4$ 寻优曲线对比图6 函数 $f_5$ 寻优曲线对比图7 函数 $f_6$ 寻优曲线对比

通过对标准测试函数的仿真实验结果表明,改进后的 BA 算法在寻优精度上远优于传统的 BA 算法,由此说明改进后的 BA 算法较传统的 BA 算法在全局搜索能力上有所提高,并且

具有较好的收敛精度。

## 2 聚类分析算法

### 2.1 K-means 聚类算法

经典的 K-means 聚类算法通过在样本空间中随机选取  $K$  个点作为初始聚类中心,采用距离计算公式(一般采用欧氏距离)计算空间中其余数据对象到聚类中心的距离,通过对比将距离小的数据进行归类,进行迭代更新聚类中心的值,从而获得最优的聚类结果。

欧氏距离:向量  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  和向量  $\mathbf{y} = (y_1, y_2, \dots, y_n)$  的欧氏距离用  $d(x_i, y_i)$  表示,其定义如式(9)所示。

$$d(\mathbf{x}, \mathbf{y}) = [\sum_{i=1}^n (x_i - y_i)^2]^{1/2} \quad (9)$$

对于聚类质量的评估函数,本文拟采用类内距离和:

$$J = \sum_{j=1}^k \sum_{i=1}^{n_j} d(x_i, z_j) \quad x_i \in C_j \quad (10)$$

其中: $n_j$  是子类  $C_j$  中的样本总数; $z_j$  表示子类  $C_j$  的聚类中心;

$z_j = \frac{1}{n_j} \sum_{i=1}^{n_j} x_i$  是该子类所含样本的均值; $d(x_i, z_j)$  表示样本  $x_i$  与所属子类  $C_j$  的中心  $z_j$  的距离; $J$  表示全部类内距离和, $J$  值越小,说明聚类效果越好。经典 K-means 聚类算法流程如图 8 所示。

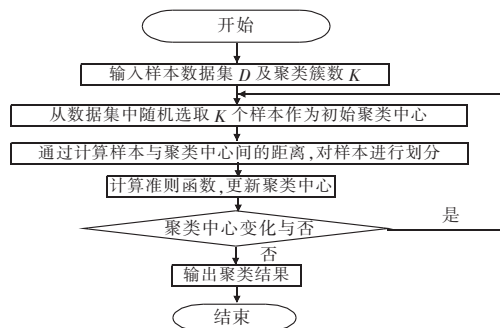


图8 经典 K-means 聚类算法流程

### 2.2 基于改进 BA 算法的 K-means 聚类

LWBA-K-means 算法主要步骤:

a) 输入样本数据,设置算法参数。蝙蝠种群大小为  $N$ ,算法最大迭代次数为  $\maxCycle$ ,脉冲发射率为  $r$ ,响度为  $A$ ,频率为  $[f_{\min}, f_{\max}]$ ,响度及脉冲发射率变化因子为  $\alpha, \gamma$ ,聚类簇数为  $K$ ,阈值为  $L$ 。

b) 初始化种群。根据读入的样本数据和聚类簇数,随机产生蝙蝠种群位置及具有最优位置的蝙蝠个体。

c) 对蝙蝠群进行一次聚类划分,每个蝙蝠代表一种聚类划分,并按照式(10)计算每个蝙蝠对应的适应度值,更新当前具有最优适应度值的蝙蝠个体。

d) 根据式(1)(2)(7)进行搜索产生新的位置,对新位置进行评估,如果  $\text{rand} > r_i$  或实验次数大于阈值  $L$ ,在当前最优位置处产生随机扰动,用扰动后的位置代替当前位置。

e) 如果  $\text{rand} > A_i$  且  $f_{\text{new}} > f_i$ ,则移动到更新后的位置,如果更新位置后的蝙蝠个体优于当前群体的最佳蝙蝠个体,则替换掉最佳蝙蝠个体,并根据式(5)调整脉冲发射率  $r$  和声音响度  $A$ 。

f) 对当前更新的位置进行评估,并找出当前最优蝙蝠个体,将新位置作为新的聚类中心,在完成搜索后可得到多个新的聚类中心,将新的聚类中心作为 K-means 的初始聚类中心进行一次 K-means 聚类,更新得到新的聚类中心,用新的聚类中心再次去更新种群。



g) 如果当前迭代次数等于最大迭代次数,结束整个算法,输出聚类结果,否则转向 c)。

### 2.3 实验仿真

为了验证本文提出的基于改进蝙蝠算法的 K-means 聚类算法性能,本文在 MATLAB 上进行实验仿真,采用国际公开测试数据集 UCI<sup>[18]</sup> 库中的数据集中的数据集进行实验。选择 UCI 数据库中著名的数据集 iris、wine 进行测试,iris 是以鸢尾花为特征的 UCI 数据集,其包含 150 个数据信息,每个数据包含 4 个属性,共分为 3 类,每类包含 50 个数据;wine 是以葡萄酒的成分分析为特征的数据集,包含 178 个数据信息,每个数据包含 13 个属性,分为 3 类,每类数据的数量为 48、59、71。

算法参数设置:最大迭代数  $\text{maxCycle} = 300$ ,种群规模  $N = 100$ ,初始响度  $A = 0.25$ ,脉冲发射率  $r = 0.5$ ,频率  $f = [0, 2]$ ,脉冲发射率增大因子  $\gamma = 0.05$ ,响度衰减因子  $\alpha = 0.95$ ,最大惯性因子  $\omega_{\max} = 0.9$ ,最小惯性因子  $\omega_{\min} = 0.2$ 。为了验证本文算法的有效性,将本文算法 LWBA-K-means 和相关聚类算法(基本 K-means,粒子群聚类(PSO))对所用于测试的数据各自运行 20 次,取 20 次运行结果的准确率的平均值作为最终准确率,结果如表 4 所示。

表 4 算法准确率对比

/%

算法	iris 数据集	wine 数据集
K-means	80.32	68.61
PSO	84.81	70.12
LWBA-K-means	91.63	72.59

根据实验结果可知,本文提出的 LWBA-K-means 算法与传统的 K-means 算法及 PSO 聚类算法相比,LWBA-K-means 聚类算法明显提升了聚类的准确率,作为一种改进的 K-means 聚类算法,所得到的聚类质量得到了改善。如图 9~11 所示。

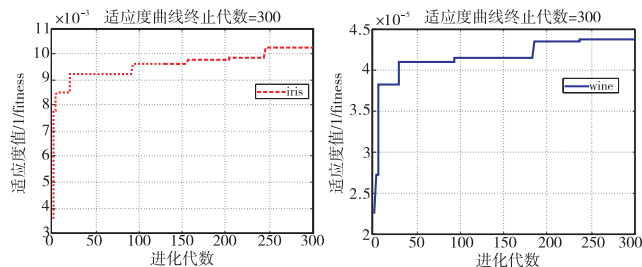


图9 LWBA-K-means 聚类iris适应度变化曲线

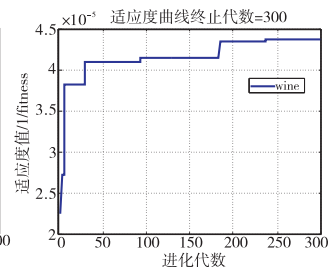


图10 LWBA-K-means 聚类wine适应度变化曲线

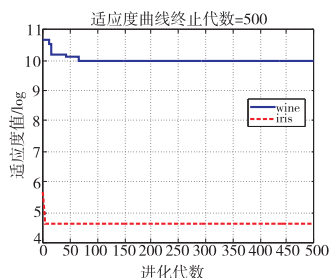


图11 LWBA-K-means 对iris及wine聚类适应度变化

为了更好地验证本文算法的有效性,通过对实验数据集分别运行 10、20、30、40、50 次的准确率的平均结果画出折线图变化规律,如图 12、13 所示。

通过对 iris 和 wine 数据集聚类准确率折线图变化趋势的分析表明,本文所提出的 LWBA-K-means 聚类算法针对不同的数据集进行聚类时随着运行次数的增加准确率逐渐趋于平缓,相较于其他聚类算法整体的效率较优。

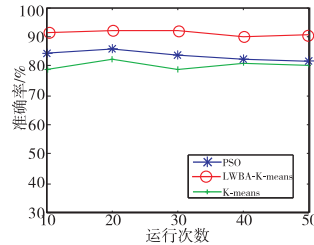


图12 对iris数据集 运行50次的准确率变化

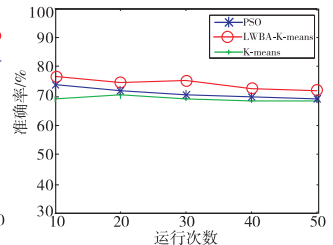


图13 对wine数据集 运行50次的聚类准确率变化

### 3 结束语

本文将 Lévy 飞行模式引入蝙蝠算法,对蝙蝠的速度和位置更新公式进行改进,提升了算法的全局搜索能力的同时使其搜索模式更趋近于现实生活中蝙蝠的运动模式;受粒子群算法的启发,将惯性权重因子引入蝙蝠算法,从而达到平衡算法全局寻优与局部寻优的能力;同时通过将 limit 阈值的概念引入 BA 算法,提高了算法跳出局部极值的能力。通过对六个标准测试函数的实验仿真,验证了所改进 BA 算法的有效性。最后,将改进后的 BA 算法同经典 K-means 聚类算法进行结合,提出了一种基于改进 BA 算法的 K-means 聚类。实验表明,本文算法相较于传统 K-means 和 PSO 聚类算法优化了算法聚类质量,且在算法的聚类效率和准确率都有所提高。

#### 参考文献:

- [1] Han Jiawei, Kamber M. 数据挖掘:概念与技术[M]. 北京:机械工业出版社,2012:288-295.
- [2] 谢娟英,王艳娥. 最小方差优化初始聚类中心的 K-means 算法[J]. 计算机工程,2014,40(8):205-211.
- [3] 吕少娟,张桂珠. 一种融合 K-means 算法和人工鱼群算法的聚类方法[J]. 计算机应用与软件,2015,32(9):240-243.
- [4] Yang Xinshe. A new metaheuristic bat-inspired algorithm[M]//Nature Inspired Cooperative Strategies for Optimization. Berlin:Springer, 2010:65-74.
- [5] Yang Xinshe, Gandomi A H. Bat algorithm: a novel approach for global engineering optimization[J]. Engineering Computations, 2012, 29(5):464-483.
- [6] 盛孟龙,贺兴时,王慧敏. 一种改进的自适应变异蝙蝠算法[J]. 计算机技术与发展,2014,24(10):131-134.
- [7] 李煜,马良. 新型全局优化蝙蝠算法[J]. 计算机科学,2013,40(9):225-229.
- [8] Ghaemi M, Zabihinpour Z, Asgari Y. Computer simulation study of the Levy flight process[J]. Physica A: Statistical Mechanics and Its Applications, 2009, 388(8):1509-1514.
- [9] 刘长平,叶春明. 具有 Lévy 飞行特征的蝙蝠算法[J]. 智能系统学报,2013,8(3):240-246.
- [10] 李枝勇,马良,张惠珍. 0-1 规划问题的元胞蝙蝠算法[J]. 计算机应用研究,2013,30(10):2903-2906.
- [11] 谢健,周永权,陈欢. 一种基于 Lévy 飞行轨迹的蝙蝠算法[J]. 模式识别与人工智能,2013,26(9):829-837.
- [12] Mandelbrot B B, Wheeler J A. The fractal geometry of nature[J]. Quarterly Review of Biology, 1983, 51:286-287.
- [13] Shi Yuhui, Eberhart R. A modified particle swarm optimizer[C]//Proc of IEEE International Conference on Evolutionary Computation. Anchorage, AK:IEEE Press, 1998:69-73.
- [14] 沈艳,余冬华,王浩雷. 粒子群 K-means 聚类算法的改进[J]. 计算机工程与应用,2014,50(21):125-128.
- [15] 谢娟英,鲁肖肖,屈亚楠,等. 粒计算优化初始聚类中心的 K-means 聚类算法[J]. 计算机科学与探索,2015,9(5):611-620.
- [16] 程国,丁正生. 一种求解旅行商问题的改进人工蜂群算法[J]. 河南科学,2015,33(6):883-887.
- [17] 刘川川,丁海军. 一种基于改进人工蜂群的 K-means 聚类算法[J]. 微处理,2016,37(2):47-50.
- [18] University of California, Irvine. UCI machine learning repository [DB/OL]. [2013-06-19]. <http://archive.ics.uci.edu/ml>.