

融合多策略改进的麻雀搜索算法及其在疾病预测中的应用^①



王 婷, 孙金泽, 赵 倩, 荆长强

(临沂大学 信息科学与工程学院, 临沂 276000)

通信作者: 荆长强, E-mail: jingchangqiang@lyu.edu.cn

摘 要: 针对麻雀搜索算法 SSA 求解问题时收敛较快, 容易陷入局部最优解的问题, 本文提出一种融合多策略改进的麻雀搜索算法 LCSSA. 首先通过采用非线性递减权重和 Levy 飞行策略共同改进发现者位置更新公式以提高算法的全局搜索能力与及时跳出局部最优解的能力; 其次, 通过引入柯西变异对跟随者位置更新, 即对最优解进行更新扰动. 本文选取 4 个对比算法在 12 个标准基准函数上进行对比实验验证. 实验结果表明, 改进后的算法在收敛速度以及稳定性方面取得了有效性的提升. 将 LCSSA 应用于疾病预测中, 算法在 4 种慢性疾病数据集中表现较好, 与对比算法相比具有更高的预测精度.

关键词: 麻雀搜索算法; LSSVM 模型; 疾病诊断; Levy 飞行; 柯西变异

引用格式: 王婷, 孙金泽, 赵倩, 荆长强. 融合多策略改进的麻雀搜索算法及其在疾病预测中的应用. 计算机系统应用, 2025, 34(4): 239–247. <http://www.c-s-a.org.cn/1003-3254/9851.html>

Improved Sparrow Search Algorithm Based on Multi-strategy Fusion and Application in Disease Prediction

WANG Ting, SUN Jin-Ze, ZHAO Qian, JING Chang-Qiang

(School of Information Science and Engineering, Linyi University, Linyi 276000, China)

Abstract: To address the fast convergence that leads to a tendency to local optimal solutions of the sparrow search algorithm SSA when solving problems, this study proposes a sparrow search algorithm incorporating multi-strategy improvement (LCSSA). Firstly, the ability of global searching and to jump out of local optimal solutions is enhanced by introducing nonlinear decreasing weights and Levy flight strategy to jointly improve the discoverer position updating formula. Secondly, Cauchy mutation is introduced to update the positions of the followers, that is, the optimal solution is updated and perturbed. The study selects four comparison algorithms on 12 benchmark functions for comparative experiments. The experimental results show that the improved algorithm has achieved effective improvement in convergence speed and stability. In disease prediction, LCSSA has a good performance in four chronic disease datasets, showing higher prediction accuracy compared with selected algorithms.

Key words: sparrow search algorithm (SSA); LSSVM model; disease diagnosis; Levy flight; Cauchy mutation

群体智能优化算法是一种基于群体行为模式的优化算法, 通过模拟群体中的相互作用和信息交流来求解复杂问题, 基于此许多学者提出了一系列的算法, 如

蚁群算法 (ACO)、粒子群优化算法 (PSO) 等. 群体智能优化算法凭借其操作简便、结构精简、高鲁棒性等优点, 在现实世界中得到了广泛的应用.

^① 基金项目: 山东省数据开放创新应用实验室 (23924010)

收稿时间: 2024-10-11; 修改时间: 2024-11-07; 采用时间: 2024-12-11; csa 在线出版时间: 2025-03-05

CNKI 网络首发时间: 2025-03-06

麻雀搜索算法 (sparrow search algorithm, SSA) 在 2020 年由 Xue 等人首次提出, 现成功应用于多个领域之中^[1,2]. 该算法在求解问题时存在容易陷入局部最优等问题, 因此许多学者提出了不同的策略对其进行改进研究. Ma 等人^[3]在动态进化策略中, 设计了相似扰动函数, 并引入三角形相似理论来提高探索能力. Li 等人^[4]提出通过反向学习、余弦惯量权重和 Levy 飞行策略结合改进 SSA, 并验证了改进后的算法在优化方面的优越性. Li 等人^[5]依次集成 5 种策略, 实验结果得出正弦余弦算法和 Levy 飞行策略对 SSA 的改善影响更大. Liu 等人^[6]使用 Cauchy 反向学习初始化麻雀种群, 并采用正弦余弦算法更新生产者位置, 最后将改进的 SSA 与动态窗口方法 (DWA) 相结合. 回立川等人^[7]采用邻域重心反向学习策略进行初始化, 引入 Levy 飞行策略与自适应权重进行位置更新. 刘博等人^[8]通过 Cubic 混沌映射初始化种群, 融合鸟群算法改进发现者位置更新. 乐明皓等人^[9]将 Sobol 序列、随机反向学习策略、螺旋觅食策略以及柯西变异等多策略融合, 将其应用于 AGV 路径规划中. 李江华等人^[10]引入了精英反向学习策略初始化种群.

上述文献对麻雀搜索算法有一定的改进, 但仍然存在开发能力不足等缺点, 本文提出一种融合多种策略改进的麻雀搜索算法 (LCSSA), 该算法在发现者位置更新时采用非线性递减权重与 Levy 飞行相结合的方法提高发现者的全局搜索能力, 同时引进柯西变异策略对跟随者位置更新, 帮助算法在陷入停滞状态时有效脱离. 本文通过对 12 个基准函数进行仿真实验, 实验结果验证了该改进策略的可行性与有效性.

疾病的早期预测和准确诊断在医学领域扮演着举足轻重的角色. 及时发现疾病迹象使患者能够更早期地接受治疗, 提高患者的康复率和生活质量. 目前, 关于疾病分类预测的研究已经取得了一定的进展^[11-13]. 通过利用机器学习等技术, 能够分析患者的数据, 找出其中的规律和特征, 实现疾病的早期预测和准确诊断. 因此, 本文将改进的麻雀搜索算法优化 LSSVM 模型参数, 实现 LSSVM 模型超参数的自动化调整, 本文将通过详尽的实验验证和性能分析, 全面评估该模型的有效性, 并期望为慢性疾病的早期预警和治疗提供创新且实用的解决方案.

1 SSA 优化算法

麻雀搜索算法 (SSA) 主要是受麻雀的觅食行为和

反捕食行为的启发. 麻雀在觅食过程中, 会展现出发现者和加入者两种行为策略, 同时还会对周围环境保持警惕, 以应对潜在的捕食者.

发现者又叫捕食者, 在种群中负责寻找食物, 并为整个麻雀种群提供觅食区域和方向. 当环境安全时 (即预警值 R_2 小于安全值 ST), 发现者会进行广泛的搜索; 当环境危险时 (即 R_2 大于等于 ST), 发现者会引导整个群体逃离到安全区域. 发现者的更新公式为:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \cdot \exp\left(\frac{-i}{\alpha \cdot iter_{\max}}\right), & R_2 < ST \\ X_{i,j}^t + Q \cdot L, & R_2 \geq ST \end{cases} \quad (1)$$

其中, $X_{i,j}^t$ 代表第 i 个麻雀在第 j 维度下的位置信息, t 代表当前迭代次数; $iter_{\max}$ 为最大迭代次数; α 与 Q 为 $[0, 1]$ 与服从正态分布的随机数; L 为单位矩阵.

加入者又称追随者, 利用发现者来获取食物. 它们会跟随发现者前往食物源, 并可能根据觅食过程中的情况转变为发现者. 加入者的位置更新则依赖于当前发现者的最优位置和全局最差位置. 加入者的位置公式为:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{\text{worst}}^t - X_{i,j}^t}{t^2}\right), & \text{if } i > \frac{n}{2} \\ X_p^{t+1} + |X_{i,j}^t - X_p^{t+1}| \cdot A^+ \cdot L, & \text{otherwise} \end{cases} \quad (2)$$

其中, X_{worst} 与 X_p 分别为全局最差位置与目前全局最优位置; $A^+ = A^T \cdot (A \cdot A^T)^{-1}$.

警戒者负责监视周围环境, 以防止潜在威胁 (如捕食者) 的入侵. 当发现危险时, 警戒者会立即发出警报, 引导整个群体逃离危险区域. 警戒者的位置公式为:

$$X_{i,j}^{t+1} = \begin{cases} X_{\text{best}}^t + \beta \cdot |X_{i,j}^t - X_{\text{best}}^t|, & \text{if } f_i > f_g \\ X_{i,j}^t + K \cdot \left(\frac{|X_{i,j}^t - X_{\text{worst}}^t|}{(f_i - f_w) + \varepsilon}\right), & \text{if } f_i = f_g \end{cases} \quad (3)$$

其中, X_{best} 为当前全局最优位置; K 与 β 分别为 $[-1, 1]$ 与服从正态分布的随机数; f_i 表示当前麻雀个体的适应值; f_g 表示全局最佳适应度值; f_w 表示全局最差适应度值; ε 是防止除 0 的常数.

2 改进的麻雀搜索算法

2.1 非线性递减权重更新发现者位置

SSA 在迭代过程中, 发现者的位置更新往往采用固定的权重, 这可能导致算法在搜索过程中过早收敛于局部最优解, 而非全局最优解. 为了克服这一缺陷,

引入非线性递减权重的策略来优化发现者的位置更新机制,使其能够更广泛地搜索解空间,避免陷入局部最优的陷阱。具体地,引入非线性递减权重后的发现者位置更新方程表示如下:

$$X_{i,j}^{t+1} = \begin{cases} \omega \cdot X_{i,j}^t \cdot \exp\left(\frac{-i}{\alpha \cdot \text{iter}_{\max}}\right), & R_2 < ST \\ \omega \cdot X_{i,j}^t + Q \cdot L, & R_2 \geq ST \end{cases} \quad (4)$$

$$\omega = \left(\frac{\text{iter}_{\max} - t + 1}{\text{iter}_{\max}} \right)^t \quad (5)$$

其中, w 为更新权重, t 表示当前迭代次数。随着迭代次数的增加,权重逐渐非线性递减,即在搜索后期,发现者的探索范围逐渐缩小,转向更为精细的局部搜索,从而能够更准确地定位到全局最优解附近。

2.2 Levy 飞行策略

Levy 飞行以其长距离跳跃与短距离精细搜索相结合的特性,提高发现者的全局搜索能力^[14]。通过在位置更新公式中引入非线性递减权重策略,以有效提高收敛效果。然而在应对高维复杂问题时,算法仍可能陷入局部最优。融入 Levy 飞行项,算法能够促使发现者以一定概率执行大范围的跳跃,从而跳出局部最优区域,改善了其解的随机性和多样性,Levy 飞行位置更新公式如下:

$$x_i'(t) = x_i(t) + l \oplus \text{levy}(\lambda) \quad (6)$$

其中, l 表示步长控制参数, $l=0.01(x_i(t)-x_p)$, x_p 表示目前最优解, $\text{levy}(\lambda)$ 表示搜索路径,满足 $\text{levy} \sim u = t^{-\lambda}$, $1 < \lambda \leq 3$, 利用 Mantegna 算法得到 Levy 飞行路径:

$$\begin{cases} s = \frac{\mu}{|v|^{1/\gamma}} \\ \mu \sim N(0, \sigma_\mu^2), v \sim N(0, \sigma_v^2) \\ \sigma_\mu = \left\{ \frac{\Gamma(1+\gamma) \sin(\pi\gamma/2)}{\gamma \cdot \Gamma[(1+\gamma)/2] \cdot 2^{(1+\gamma)/2}} \right\}^{1/\gamma} \end{cases} \quad (7)$$

Levy 飞行机制和非线性递减权重结合,使得算法在求解过程中能够产生高质量解,很大程度上增强算法的搜索能力。Levy 飞行示意图如图 1 所示。

2.3 柯西变异策略更新跟随者位置

SSA 算法中,跟随者通常围绕最优的发现者进行觅食,以期找到更好的优化问题的解。引入柯西变异(Cauchy mutation)^[15]策略对跟随者的位置更新机制进行改进。新的加入者位置更新表示如下:

$$x_{i,j}^{t+1} = X_{\text{best}}(t) + \text{cauchy}(0, 1) \oplus X_{\text{best}}(t) \quad (8)$$

其中, $\text{cauchy}(0, 1)$ 为标准柯西分布函数:

$$f(x) = \frac{1}{\pi} \left(\frac{1}{x^2 + 1} \right), -\infty < x < \infty \quad (9)$$

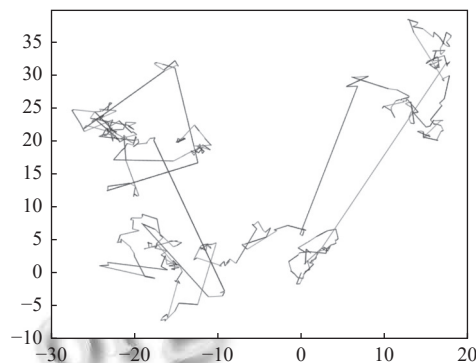


图 1 Levy 飞行示意图

引入柯西分布来生成变异项,增强算法的多样性和全局探索能力,从而允许算法在小范围精细搜索和大幅度跳跃探索之间灵活切换。

2.4 LCSSA 算法步骤

融合多策略改进的麻雀搜索算法 LCSSA 算法的伪代码如算法 1 所示。

算法 1. LCSSA 算法

输入: 种群规模 pop 、最大迭代次数 Max_iter 、 ST 、 PD 、 SD 、搜索空间上下界 lb 、 ub 。

输出: 更新最优适应度值 $Best_Fit$ 与最佳位置 $Best_Pos$ 。

1. $population = \text{Initialize_pop}(pop, max, lb, ub)$ // 初始化种群
2. for $iter$ from 1 to Max_iter do
3. for individual in discoverers do
4. 使用式 (4) 与式 (6) 计算更新发现者位置
5. end for
6. for individual in followers do
7. 使用式 (8) 计算更新跟随者位置
8. end for
9. for individual in scouts do
10. 使用式 (3) 计算更新警戒者位置
11. end for
12. $new_fitness = \text{Evaluate_Fitness}(\text{individual.position})$
13. end for
14. return $Best_Fit, Best_Pos$

3 算法性能测试

3.1 实验设置与测试函数

为验证 LCSSA 的性能表现,本文选取了 PSO^[16]、GWO^[17]、CPO^[18]、SSA 这 4 种具有代表性的算法作为参照对象。实验中,统一设置种群数量为 50,最大迭代次数为 500,以确保所有算法在相同的条件下进行比

较. 同时, 每种算法均独立运行 30 次, 以获取更为稳定和可靠的实验结果. 每种算法的初始参数设定如表 1 所示. 此外, 实验采用了 12 个经典测试函数作为评估基准, 其中 F1-F7 为单峰函数, F9-F13 为多峰函数, 并分别对测试函数的维度设置为 30、50、100 进行实验, 基准测试函数如表 2 所示.

表 1 算法参数设置

算法	参数
PSO	$w_{Max}=0.8, w_{Min}=0.2, c1=c2=1.5$
GWO	$a \in [0, 2]$, 从 2 到 0 线性递减
CPO	$\alpha=0.2, N_{min}=40, T_f=0.8, T=2$
SSA	$PD=0.2, SD=0.2, ST=0.8$
LCSSA	$PD=0.2, SD=0.2, ST=0.8$

本实验设备采用 AMD Ryzen 97945HX 处理器, RAM 16.0 GB, Windows 11 64 位操作系统, 实验平台

为 MatlabR2024a.

表 2 基准测试函数

编号	测试函数	维数	搜索范围	最优值
F1	Sphere	Dim	$[-100, 100]$	0
F2	Schwefel 2.22	Dim	$[-10, 10]$	0
F3	Schwefel 1.2	Dim	$[-100, 100]$	0
F4	Schwefel 2.21	Dim	$[-100, 100]$	0
F5	Rosenbrock	Dim	$[-30, 30]$	0
F6	Step	Dim	$[-100, 100]$	0
F7	Quartic	Dim	$[-1.28, 1.28]$	0
F9	Rastrigin	Dim	$[-5.12, 5.12]$	0
F10	Ackley	Dim	$[-32, 32]$	0
F11	Griewank	Dim	$[-600, 600]$	0
F12	Penalized1	Dim	$[-50, 50]$	0
F13	Penalized2	Dim	$[-50, 50]$	0

3.2 算法性能结果对比分析

算法性能结果如表 3 所示.

表 3 3 种维度下测试函数运行结果

函数	算法	Dim=30			Dim=50			Dim=100		
		最优值	平均值	标准差	最优值	平均值	标准差	最优值	平均值	标准差
F1	PSO	9.24E-03	9.85E-02	8.02E-02	5.51E+00	2.06E+01	9.55E+00	6.61E+02	1.01E+03	2.58E+02
	GWO	9.08E-35	1.85E-33	3.08E-33	1.94E-25	8.53E-24	1.31E-23	3.46E-16	3.61E-15	2.18E-15
	CPO	4.15E-76	2.19E-41	1.20E-40	5.97E-82	2.49E-41	1.16E-40	3.16E-63	1.64E-38	8.99E-38
	SSA	0.00E+00	2.32E-109	1.27E-108	0.00E+00	6.63E-110	3.63E-109	0.00E+00	1.13E-145	6.17E-145
	LCSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F2	PSO	1.14E-02	3.83E-02	2.32E-02	5.79E-01	3.22E+00	4.85E+00	1.41E+01	3.38E+01	1.16E+01
	GWO	8.02E-21	6.20E-20	5.03E-20	4.11E-15	1.55E-14	6.66E-15	7.21E-10	1.66E-09	5.61E-10
	CPO	8.27E-37	7.46E-24	2.82E-23	3.42E-35	4.82E-21	2.46E-20	1.30E-33	5.02E-21	2.06E-20
	SSA	0.00E+00	2.92E-80	1.57E-79	0.00E+00	8.39E-79	4.60E-78	0.00E+00	2.79E-74	1.52E-73
	LCSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F3	PSO	3.60E+02	1.35E+03	1.37E+03	5.44E+03	1.07E+04	3.20E+03	4.90E+04	7.33E+04	1.89E+04
	GWO	3.05E-11	1.91E-06	1.04E-05	1.44E-05	7.17E-03	1.90E-02	7.89E-01	1.14E+02	1.61E+02
	CPO	3.77E-66	1.87E-38	7.43E-38	0.00E+00	9.06E-37	4.43E-36	3.53E-69	1.15E-33	5.92E-33
	SSA	0.00E+00	1.98E-98	9.57E-98	0.00E+00	1.89E-80	1.04E-79	0.00E+00	9.79E-76	5.36E-75
	LCSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F4	PSO	4.12E+00	5.56E+00	9.43E-01	1.28E+01	1.65E+01	2.13E+00	3.12E+01	3.67E+01	3.65E+00
	GWO	5.27E-09	2.07E-08	1.45E-08	2.96E-06	4.15E-05	7.13E-05	1.59E-02	1.80E-01	2.24E-01
	CPO	1.85E-37	2.62E-21	1.16E-20	2.72E-36	3.01E-21	1.34E-20	2.93E-36	5.83E-22	2.38E-21
	SSA	0.00E+00	4.88E-64	2.63E-63	3.43E-196	1.15E-49	6.31E-49	0.00E+00	2.71E-58	1.48E-57
	LCSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F5	PSO	2.96E+01	3.11E+03	1.64E+04	5.88E+02	2.05E+03	1.07E+03	7.04E+04	2.34E+05	8.15E+04
	GWO	2.56E+01	2.67E+01	6.92E-01	4.60E+01	4.69E+01	6.29E-01	9.57E+01	9.77E+01	9.15E-01
	CPO	2.48E+01	2.53E+01	2.69E-01	4.54E+01	4.58E+01	3.02E-01	9.57E+01	9.62E+01	2.23E-01
	SSA	2.18E-03	7.68E-02	1.37E-01	1.78E-02	7.93E-01	1.46E+00	1.05E-02	2.98E+00	6.77E+00
	LCSSA	1.97E-06	1.01E-03	1.23E-03	5.71E-07	2.21E-03	5.33E-03	2.16E-05	1.24E-02	4.47E-02
F6	PSO	9.93E-03	8.57E-02	8.98E-02	5.33E-03	1.84E+01	1.12E+01	4.47E-02	1.38E+03	1.90E+03
	GWO	3.26E-05	4.18E-01	3.41E-01	7.56E-01	1.64E+00	4.40E-01	5.76E+00	8.19E+00	1.04E+00
	CPO	1.70E-06	6.37E-06	3.88E-06	3.27E-04	1.17E-03	5.58E-04	7.08E-02	1.34E-01	4.15E-02
	SSA	1.29E-03	3.53E-03	1.92E-03	1.14E-02	2.55E-02	8.88E-03	1.04E-01	1.68E-01	3.31E-02
	LCSSA	7.06E-08	3.33E-06	4.04E-06	1.32E-08	2.88E-05	3.95E-05	7.63E-08	1.06E-04	2.04E-04
F7	PSO	1.08E-02	3.17E-02	8.81E-03	7.51E-02	1.47E-01	4.43E-02	8.26E-01	1.55E+00	1.97E+00
	GWO	3.13E-04	1.08E-03	6.89E-04	2.94E-04	1.78E-03	9.19E-04	1.36E-03	3.82E-03	1.50E-03
	CPO	1.66E-04	1.53E-03	1.03E-03	1.96E-04	1.43E-03	1.11E-03	1.31E-04	1.74E-03	1.07E-03

表 3 3 种维度下测试函数运行结果 (续)

函数	算法	Dim=30			Dim=50			Dim=100		
		最优值	平均值	标准差	最优值	平均值	标准差	最优值	平均值	标准差
F7	SSA	3.23E-06	2.97E-04	2.98E-04	2.71E-05	2.84E-04	2.73E-04	1.72E-05	2.18E-04	1.54E-04
	LCSSA	3.32E-07	4.56E-05	4.71E-05	4.57E-07	5.01E-05	5.16E-05	4.88E-06	4.04E-05	3.04E-05
F9	PSO	2.56E+01	4.77E+01	1.48E+01	6.91E+01	1.06E+02	2.51E+01	2.74E+02	3.58E+02	5.10E+01
	GWO	0.00E+00	1.90E+00	3.08E+00	1.14E-13	4.28E+00	7.05E+00	3.07E-12	7.39E+00	7.41E+00
	CPO	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	SSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	LCSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F10	PSO	3.11E-02	1.68E-01	2.13E-01	1.35E+00	2.35E+00	4.50E-01	5.18E+00	5.92E+00	4.89E-01
	GWO	3.60E-14	4.41E-14	6.27E-15	2.78E-13	4.73E-13	1.84E-13	2.91E-09	6.52E-09	2.23E-09
	CPO	4.44E-16	6.81E-16	9.01E-16	4.44E-16	5.63E-16	6.49E-16	4.44E-16	5.63E-16	6.49E-16
	SSA	4.44E-16	4.44E-16	0.00E+00	4.44E-16	4.44E-16	0.00E+00	4.44E-16	4.44E-16	0.00E+00
	LCSSA	4.44E-16	4.44E-16	0.00E+00	4.44E-16	4.44E-16	0.00E+00	4.44E-16	4.44E-16	0.00E+00
F11	PSO	3.98E-02	1.51E-01	8.39E-02	1.04E+00	1.16E+00	1.21E-01	6.65E+00	1.01E+01	2.26E+00
	GWO	0.00E+00	4.10E-03	9.05E-03	0.00E+00	2.84E-03	6.97E-03	1.11E-15	1.94E-03	6.12E-03
	CPO	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	SSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
	LCSSA	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F12	PSO	9.34E-04	7.68E-02	9.63E-02	6.60E-01	3.96E+00	2.18E+00	3.29E+01	9.40E+02	1.61E+03
	GWO	5.32E-06	3.00E-02	1.93E-02	3.72E-02	7.55E-02	2.66E-02	1.44E-01	2.13E-01	5.81E-02
	CPO	4.68E-08	1.87E-07	1.14E-07	6.49E-06	1.52E-05	4.61E-06	5.48E-04	9.55E-04	2.21E-04
	SSA	4.67E-05	1.14E-03	5.83E-04	1.50E-03	2.68E-03	5.93E-04	2.86E-03	3.80E-03	5.92E-04
	LCSSA	2.45E-12	9.00E-08	1.22E-07	1.43E-10	8.33E-07	1.09E-06	6.00E-10	1.24E-06	1.92E-06
F13	PSO	1.36E-02	1.26E-01	1.32E-01	1.72E+01	4.17E+01	1.90E+01	4.13E+03	5.25E+04	4.35E+04
	GWO	1.32E-01	4.02E-01	2.13E-01	9.28E-01	1.61E+00	3.69E-01	4.87E+00	6.04E+00	5.40E-01
	CPO	9.63E-07	7.36E-04	2.79E-03	1.77E-04	9.89E-04	2.01E-03	5.46E-02	1.01E-01	3.27E-02
	SSA	1.55E-04	2.45E-01	1.98E-01	7.92E-04	3.96E-01	2.48E-01	8.64E-02	5.06E-01	4.72E-01
	LCSSA	1.27E-09	1.87E-06	2.94E-06	5.45E-08	2.38E-05	3.04E-05	1.67E-07	5.51E-05	8.72E-05

通过表 3 的数据可以得出, LCSSA 算法在单峰测试函数中, 寻优能力均强于对比算法, 特别是在 F1-F4 函数上寻优结果取得最优解. 对于多峰函数, 除 F12 的寻优结果在标准差上略逊 CPO 算法, 在其他函数的寻优结果上均强于对比算法, 且在 F9、F11 函数上寻优结果取得最优解. 进一步地从不同维度上分析, LCSSA 在 50 维与 100 维度上的 12 种测试函数中的寻优结果

均优于对比算法, 仅在 30 维度的 F12 函数上表现次于 CPO 算法. 总体来看, LCSSA 算法具有更强的搜索能力与较高的稳定性.

3.3 算法收敛曲线图

为直观展示 LCSSA 算法的动态收敛性能, 图 2 展示了在 30 维下, 各算法在 12 个测试函数上的收敛曲线对比图.

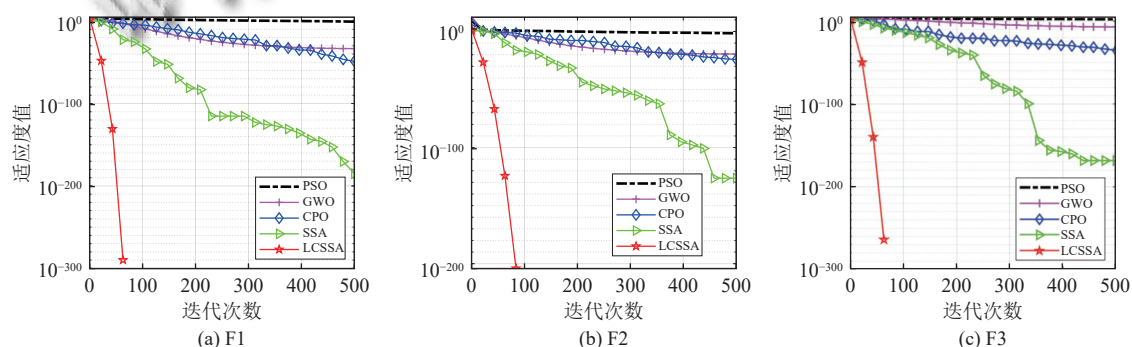


图 2 在 12 个测试函数下的收敛曲线

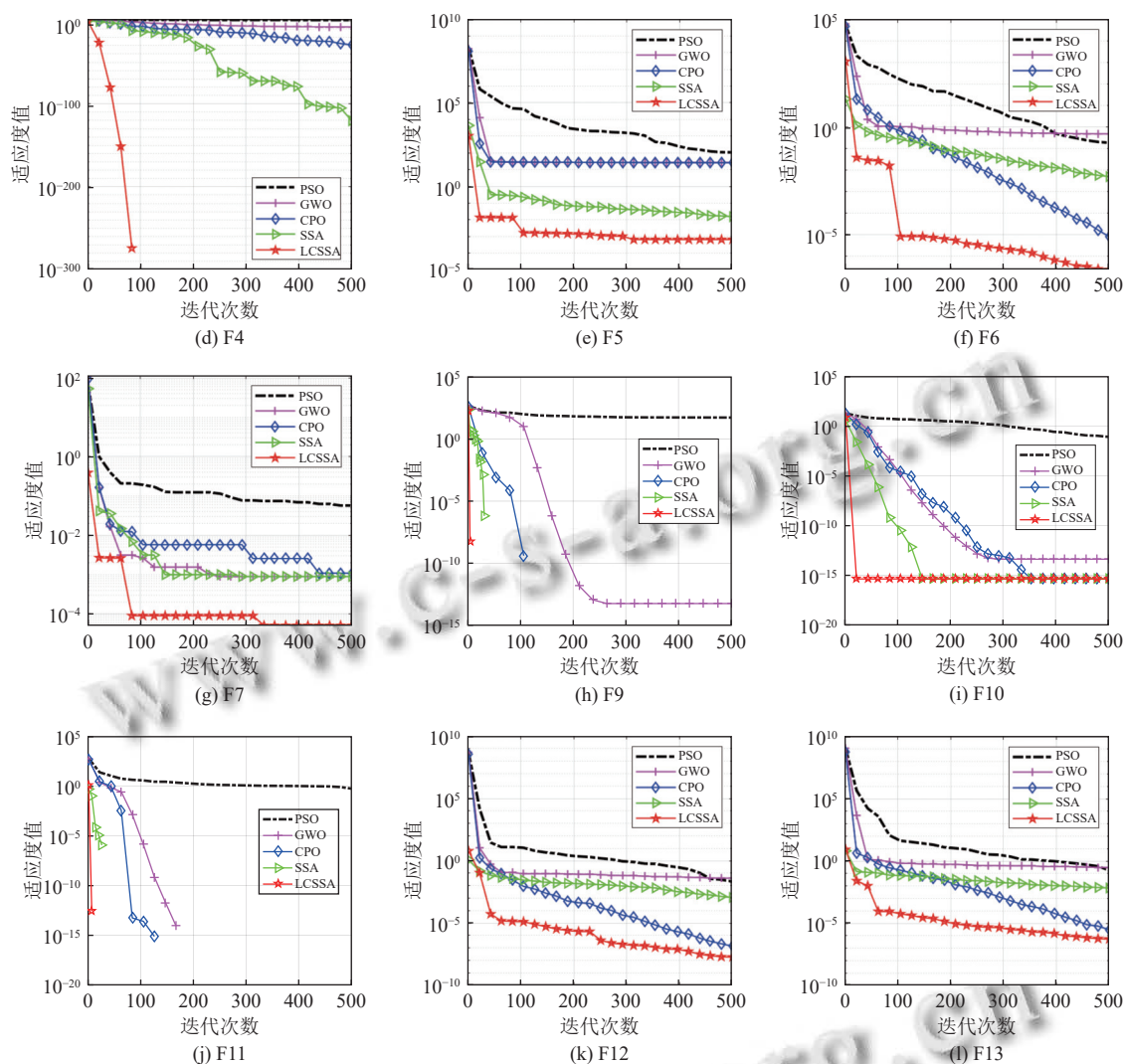


图2 在12个测试函数下的收敛曲线(续)

从收敛曲线图观察到,无论是在低维基准测试函数还是高维测试函数下,LCSSA算法都展现出了更为优越的收敛性能.图中展示LCSSA的收敛曲线都处于图像最下方,寻优过程中达到的数值越低,代表寻优精度越高.其次,在收敛速度方面,LCSSA的收敛曲线往往先于其他算法出现拐点,求解速度更快.另外,LCSSA算法在F6、F7的收敛曲线呈现出阶梯波动式下降的特点,表明了柯西变异策略在算法中的有效应用.柯西变异策略能够帮助算法在陷入停滞状态时有效脱离,从而在全局范围内继续搜寻更优的解.

4 LCSSA优化LSSVM疾病预测

4.1 LSSVM算法

LSSVM的基本思想是通过非线性映射将原始数

据空间映射到高维特征空间,并在该特征空间中寻找一个最优超平面,使得不同类别的样本点能够被最大程度地分隔开^[19].与传统SVM^[20]不同,LSSVM使用最小二乘线性系统作为损失函数,通过求解线性方程组来找到最优超平面的参数.

在LSSVM算法中,核函数决定了数据从原始空间到高维空间的映射方式,影响着模型的分类边界和决策面的形成.惩罚因子能够帮助模型在追求训练精度和保持泛化能力之间找到最佳的平衡点,避免过拟合或欠拟合的现象.因此,在构建LSSVM模型时,通过LCSSA的智能搜索和优化机制,来找到更适合的核函数和惩罚因子的组合,从而能够提升LSSVM模型的分类效果和泛化能力.

4.2 LCSSA-LSSVM模型建立

在LCSSA-LSSVM模型中,LCSSA优化算法的目

标是自动搜索最佳的惩罚系数 c 和核函数参数 g , 以提高 LSSVM 的分类性能. 通过优化这两个关键参数, 可以减少模型的过拟合或欠拟合现象. 算法 2 是基于 LCSSA-LSSVM 的疾病诊断分类预测算法的具体实施步骤.

算法 2. LCSSA-LSSVM 算法

- 1) 初始化. 设定参数范围: c 和 g 的搜索范围为 $[0.01, 100]$; 初始化麻雀种群: 种群规模 $pop=10$, 最大迭代次数 $Max_iter=50$, 预警值 $ST=0.8$, 发现者比例 $PD=0.2$, 警戒者比例 $SD=0.2$; 随机生成麻雀的初始位置;
- 2) 定义适应度函数. 对于每个麻雀个体, 使用 LSSVM 模型进行五折交叉验证得到适应度值, 计算平均错误率作为算法的目标值;
- 3) 迭代优化. 对于 $i=1$ 到 Max_iter : 使用 LCSSA 算法的公式更新麻雀个体的位置; 计算更新后麻雀的适应度值; 比较当前迭代的适应度值与历史最优适应度值, 更新最优适应度值和最优解.
- 4) 输出结果. 输出最优 c 和 g 参数值; 使用最优参数训练 LSSVM 模型, 并在测试集上评估分类性能.

5 实验设计与结果分析

5.1 实验数据集

为了验证本文所提出算法的实际效能, 实验数据选取 UCI 主页上的 2 个公共数据集以及 Kaggle 平台上的 2 个公共数据集进行对比实验, 具体包括 Parkinsons、WDBC、Diabetes、和 SAHeart. 实验数据集的相关信息如表 4 所示.

表 4 实验数据集

序号	数据集	特征数	实例数
1	Parkinsons	23	197
2	WDBC	30	569
3	Diabetes	9	768
4	SAHeart	10	462

表 4 中, Parkinsons 数据集旨在区分帕金森病患者与健康个体; WDBC 数据集源自南斯拉夫卢布尔雅那大学医疗中心肿瘤研究所, 用于预测乳腺癌的良恶性;

Diabetes 数据集记录了美国亚利桑那州地区糖尿病患者的相关信息; SAHeart 数据集是在南非收集的一组有关心脏病风险因素的数据. 这些多样化的数据集涵盖了医学领域的 4 个关键疾病方向, 确保了实验结果的全面性和有效性.

5.2 实验设置

为了全面验证本文提出的 LCSSA-LSSVM 分类预测模型在疾病诊断领域的有效性, 本文选取了 LSSVM、PSO-LSSVM、GWO-LSSVM、CPO-LSSVM、SSA-LSSVM 这 4 个具有代表性的分类预测算法进行对比实验. 实验采用统一的实验设置在 4 个公开数据集上进行: 将 70% 数据用作训练集, 剩余 30% 的数据用作测试集, 样本数据集进行归一化处理, 同时, 为了确保实验结果的稳健性, 采取独立执行 10 次实验验证的方法, 取其平均值作为最终的实验结果.

5.3 实验结果与应用分析

文本通过实验对比不同算法在 4 个数据集上的性能表现. 图 3 中展示了各对比算法在 4 个数据集上的准确率预测结果对比情况, 为了更清晰地呈现对比效果, 我们将 10 次实验的准确率结果按照从小到大的顺序排序并进行绘图.

从图 3 中可以观察到, 在 4 个数据集中使用 LCSSA 对 LSSVM 的参数优化效果在最低准确率上均高于其他对比算法. 从曲线的总体趋势来看, 尽管在个别几次实验中算法的预测结果略低于其他算法, 但总体上该算法的准确率曲线普遍位于其他算法之上, 显示出良好的性能表现. 此外, 该算法的准确率曲线在最低值和最高值之间的波动范围相较于其他算法更小, 进一步展示了算法的稳定性.

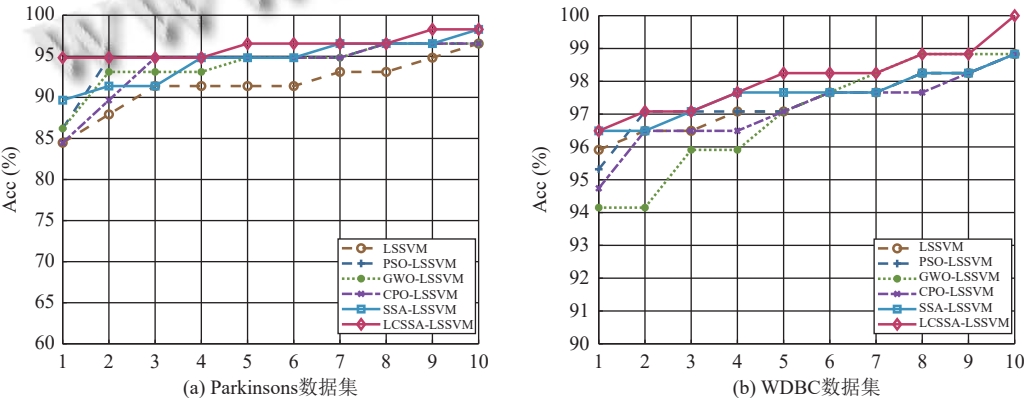


图 3 不同算法在 4 个数据集上的准确率对比

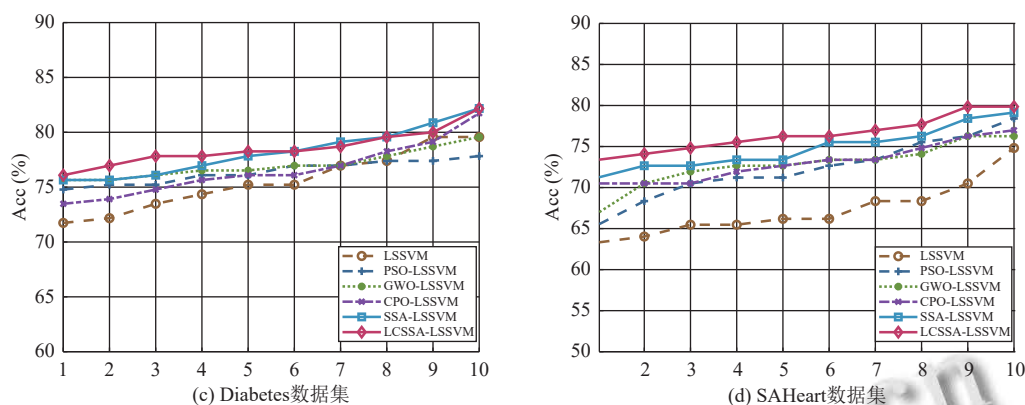


图3 不同算法在4个数据集上的准确率对比(续)

LCSSA-LSSVM 算法与其他5种对比算法在4个数据集上进行实验对比,性能结果取10次实验的平均值.表5展示了这些算法在数据集上的平均性能结果.通过对这些结果进行分析,可以更全面地评估和比较各个算法在不同数据集上的表现.

表5 各对比算法在4个数据集下的分类性能(%)

数据集	评估指标	LSSVM	PSO-LSSVM	GWO-LSSVM	CPO-LSSVM	SSA-LSSVM	LCSSA-LSSVM
Parkinsons	Acc	91.55	94.48	94.04	93.79	94.48	96.21
	Pre	85.61	91.87	88.32	89.03	90.44	93.68
	Recall	78.57	85	87.66	85	87.14	90.71
	F1	81.45	87.58	87.65	86.22	88.29	91.91
	AUC	87.13	91.25	91.87	90.8	91.98	94.33
WDBC	Acc	97.37	97.43	96.81	97.14	97.6	98.07
	Pre	98.88	98.54	99.24	98.55	99.03	99.21
	Recall	94.06	94.53	92.19	93.75	94.53	95.63
	F1	96.38	96.49	95.53	96.07	96.72	97.36
	AUC	96.7	96.85	95.88	96.46	96.99	97.58
Diabetes	Acc	75.57	76.39	77.35	76.61	78.22	78.57
	Pre	78.63	78.18	78.92	78.37	79.98	79.79
	Recall	85.93	88.53	89.15	88.67	88.88	89.93
	F1	82.1	83.02	83.69	83.17	84.16	84.55
	AUC	71.03	71.24	72.19	71.33	73.56	73.59
SAHeart	Acc	67.27	72.3	72.79	73.09	74.82	76.48
	Pre	72.29	75.21	75.64	77.32	77.23	78.49
	Recall	81.1	86.15	86.21	83.63	87.36	88.24
	F1	76.38	80.27	80.54	80.26	81.96	83.06
	AUC	61.07	66.1	66.78	68.38	69.2	71.2

实验结果表明,在Parkinsons与SAHeart两个数据集上,LCSSA-LSSVM算法在精确度(Acc)、精确率(Pre)、召回率(Recall)、F1分值以及AUC这5个关键指标上均优于其他对比算法,得到最优的预测性能.在WDBC数据集上,其精确率仅次于GWO-LSSVM算法.在Diabetes数据集上,其精确率仅次于SSA-LSSVM算法.总体上看LCSSA-LSSVM算法的分类性能优于

其他算法,这表明LCSSA在探索复杂参数空间时具有更高的效率和更好的全局搜索能力,从而能够更有效地优化LSSVM模型的性能.

6 结论

本文提出一种融合多种策略改进的麻雀搜索算法(LCSSA),并通过对12个基准函数进行仿真实验,得到以下结论.

(1) 改进算法将非线性递减权重与Levy飞行进行结合更新发现者的位置更新策略,提高了发现者的全局搜索能力;同时引进柯西变异策略对跟随者位置更新,帮助算法在陷入停滞状态时有效脱离.

(2) 算法在12个基准测试函数上展现出更好的寻优性能.LCSSA算法能够更快的得到最优解,且有不错的稳定性.

(3) 将LCSSA算法应用于慢性疾病预测中,通过其对LSSVM算法进行参数寻优,并在4个公共数据集上进行对比实验,结果证明了LCSSA-LSSVM算法在疾病诊断领域中的有效性,为疾病诊断领域提供了新的研究思路和实用工具.

参考文献

- Xue JK, Shen B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Systems Science & Control Engineering*, 2020, 8(1): 22–34.
- Xue JK, Shen B. A survey on sparrow search algorithms and their applications. *International Journal of Systems Science*, 2024, 55(4): 814–832. [doi: [10.1080/00207721.2023.2293687](https://doi.org/10.1080/00207721.2023.2293687)]
- Ma J, Hao ZY, Sun WJ. Enhancing sparrow search algorithm via multi-strategies for continuous optimization problems.

- Information Processing & Management, 2022, 59(2): 102854.
- 4 Li JJ, Wu YH. Improved sparrow search algorithm with the extreme learning machine and its application for prediction. Neural Processing Letters, 2022, 54(5): 4189–4209. [doi: [10.1007/s11063-022-10804-x](https://doi.org/10.1007/s11063-022-10804-x)]
- 5 Li J, Chen JM, Shi J. Evaluation of new sparrow search algorithms with sequential fusion of improvement strategies. Computers & Industrial Engineering, 2023, 182: 109425.
- 6 Liu LS, Liang JR, Guo KQ, *et al.* Dynamic path planning of mobile robot based on improved sparrow search algorithm. Biomimetics, 2023, 8(2): 182. [doi: [10.3390/biomimetics8020182](https://doi.org/10.3390/biomimetics8020182)]
- 7 回立川, 李瑶, 李欢欢, 等. 多策略改进的麻雀搜索算法. 辽宁工程技术大学学报 (自然科学版), 2023, 42(6): 722–732.
- 8 刘博, 李卓, 刘伟, 等. 基于改进麻雀搜索算法优化的 RSSI 定位. 传感器与微系统, 2024, 43(9): 127–130.
- 9 乐明皓, 李凌. 多策略融合的改进麻雀搜索算法及其 AGV 路径规划应用. 国外电子测量技术, 2024, 43(7): 129–139.
- 10 李江华, 王鹏晖, 李伟. 一种混合多策略改进的麻雀搜索算法. 计算机工程与科学, 2024, 46(2): 303–315. [doi: [10.3969/j.issn.1007-130X.2024.02.013](https://doi.org/10.3969/j.issn.1007-130X.2024.02.013)]
- 11 Chandrasekhar N, Peddakrishna S. Enhancing heart disease prediction accuracy through machine learning techniques and optimization. Processes, 2023, 11(4): 1210. [doi: [10.3390/pr11041210](https://doi.org/10.3390/pr11041210)]
- 12 邹琼, 吴曦, 张杨, 等. 基于麻雀搜索算法优化的 BP 神经网络模型对 2 型糖尿病肾病的预测研究. 中国全科医学, 2024, 27(8): 961–970.
- 13 陈胤玮, 王安琪, 李丰森. 群体智能优化算法在医疗领域的应用. 中国医学物理学杂志, 2024, 41(5): 646–656. [doi: [10.3969/j.issn.1005-202X.2024.05.017](https://doi.org/10.3969/j.issn.1005-202X.2024.05.017)]
- 14 Ouyang CT, Qiu YX, Zhu DL. Adaptive spiral flying sparrow search algorithm. Scientific Programming, 2021, 2021: 6505253.
- 15 李爱莲, 全凌霄, 崔桂梅, 等. 融合正余弦和柯西变异的麻雀搜索算法. 计算机工程与应用, 2022, 58(3): 91–99. [doi: [10.3778/j.issn.1002-8331.2106-0148](https://doi.org/10.3778/j.issn.1002-8331.2106-0148)]
- 16 Kennedy J, Eberhart R. Particle swarm optimization. Proceedings of the 1995 International Conference on Neural Networks (ICNN 1995). Perth: IEEE, 1995. 1942–1948.
- 17 Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. Advances in Engineering Software, 2014, 69: 46–61. [doi: [10.1016/j.advengsoft.2013.12.007](https://doi.org/10.1016/j.advengsoft.2013.12.007)]
- 18 Abdel-Basset M, Mohamed R, Abouhawwash M. Crested Porcupine Optimizer: A new nature-inspired metaheuristic. Knowledge-based Systems, 2024, 284: 111257. [doi: [10.1016/j.knosys.2023.111257](https://doi.org/10.1016/j.knosys.2023.111257)]
- 19 尹晓珊, 钟建琳, 彭宝营, 等. 基于麻雀算法优化的最小二乘支持向量机 Ti_2AlNb 切削力预测研究. 工具技术, 2023, 57(10): 63–68. [doi: [10.3969/j.issn.1000-7008.2023.10.012](https://doi.org/10.3969/j.issn.1000-7008.2023.10.012)]
- 20 Jakkula V. Tutorial on support vector machine (SVM) [Technical Report]. Pullman: School of EECS, Washington State University, 2006. <https://course.ccs.neu.edu/cs5100f11/resources/jakkula.pdf>
- (校对责编: 张重毅)