

DOI:10.19322/j.cnki.issn.1006-4710.2025.04.013

<https://xuebao.xaut.edu.cn>

引文格式: 韩子萌, 王秋萍, 王梦娜. 多策略改进黏菌算法的支持向量机参数优化[J]. 西安理工大学学报, 2025, 41(4): 584-593.

HAN Zimeng, WANG Qiuping, WANG Mengna. Enhanced slime mould algorithm with multi-strategy for parameter optimization of support vector machine[J]. Journal of Xi'an University of Technology, 2025, 41(4): 584-593.

多策略改进黏菌算法的支持向量机参数优化

韩子萌¹, 王秋萍¹, 王梦娜²

(1. 西安理工大学 理学院, 陕西 西安 710054; 2. 中国联合网络通信有限公司陕西省分公司, 陕西 西安 710075)

摘要: 支持向量机(SVM)惩罚因子和核参数的选择对分类模型的复杂度和准确性有较大影响, 基于此, 本文提出一种多策略改进黏菌算法(MESMA)对支持向量机的参数进行优化, 以提高模型分类准确率。首先, 利用佳点集初始化种群、广义正态分布搜索与自适应比例变异策略对基本的黏菌算法进行改进; 其次, 选取 CEC2017 基准测试集, 将所提算法 MESMA 与 6 种智能算法进行对比, 结果表明 MESMA 具有更好的优化性能; 最后, 利用 MESMA 算法对 SVM 参数进行优化 (MESMA-SVM), 在 UCI 数据库中选取 4 个标准数据集进行分类, 并与其它 3 种模型进行比较, 实验结果显示 MESMA-SVM 模型分类性能更好、准确率更高。

关键词: 黏菌算法; 支持向量机; 佳点集; 广义正态分布搜索; 自适应比例变异

中图分类号: TP301. 6 **文献标志码:** A **文章编号:** 1006-4710(2025)04-0584-10

Enhanced slime mould algorithm with multi-strategy for parameter optimization of support vector machine

HAN Zimeng¹, WANG Qiuping¹, WANG Mengna²

(1. Faculty of Sciences, Xi'an University of Technology, Xi'an 710054, China;

2. China United Network Communication Co., Ltd. Shaanxi Branch, Xi'an 710075, China)

Abstract: Support vector machine(SVM) parameters such as penalty parameter and kernel parameter have a significant impact on the complexity and accuracy of classification model. Based on this, this paper proposes an enhanced slime mould algorithm with multi-strategy(MESMA) to optimize the parameters for support vector machine, so that classification accuracy by the model can be promoted. Firstly, the basic slime mould algorithm is improved via strategies including good point set initialization population, generalized normal distribution search, and adaptive proportional variation strategy. Secondly, the proposed algorithm MESMA is compared with six intelligent algorithms on the CEC2017 benchmark functions. The results show that MESMA has a superior optimization performance. Finally, the SVM parameters are optimized using the MESMA algorithm (MESMA-SVM) for classification of four standard datasets selected from UCI data repository. The experimental results show that the MESMA-SVM model compared with the other three algorithms has a better classification performance and higher accuracy.

Key words: slime mould algorithm; SVM; good point set; generalized normal distribution search; adaptive proportional variation

收稿日期: 2023-10-18; 网络首发日期: 2024-06-11

网络首发地址: <https://link.cnki.net/urlid/61.1294.n.20240611.1417.006>

基金项目: 国家自然科学基金资助项目(61976176)

第一作者: 韩子萌, 女, 硕士生, 研究方向为智能计算。E-mail: 528323889@qq.com

通信作者: 王秋萍, 女, 博士, 教授, 研究方向为智能算法理论及应用、决策分析、灰色系统理论及其应用。E-mail: wqp566@sina.com

支持向量机(support vector machine, SVM)是 Cortes 等^[1]基于 VC 理论和结构风险最小化理论于 1995 年提出的一种有监督机器学习方法。近年来, SVM 已经被广泛应用于多个领域,如故障检测^[2]、医疗诊断^[3]和图像分类^[4]等。SVM 算法的设计完整且规范,在小样本数据、高维数据分类和预测方面表现良好^[5]。但是,SVM 算法中的惩罚因子和核参数对分类准确率和泛化能力有很大影响,目前还没有统一的方法来指导参数的选择。通常情况下,这些参数采用网格搜索法、梯度下降法等方法来确定,然而这些方法很容易受到局部最优的影响,同时耗时较长^[6]。群智能算法在解决参数优化问题时具有较强的并行处理能力,可以利用自身的探索与开发机制充分搜索空间中的最优解。近年来,果蝇优化算法^[7]、萤火虫算法^[8]、蝙蝠算法^[9]等群智能算法已被验证在解决 SVM 参数调优问题时具有更好的性能和更快的效率。

黏菌算法(slime mould algorithm, SMA)是 Li 等^[10]通过模拟多头绒泡菌觅食过程中的行为和形态变化,于 2020 年设计的一种新型群智能算法。黏菌算法利用生物振荡器传播波的正负反馈机制(通过自适应权重调节),能自适应地找到链接食物的最佳路径,其振荡向量可以很好地平衡算法的开发和探索能力。自 SMA 提出以来,因其简单的结构和良好的优化能力已被应用于多个领域,如光伏模型参数识别^[11]、3D 物联网定位^[12]、图像分割^[13]等。

为进一步提高 SMA 的性能,使其更适用于 SVM 的参数选择问题,本文提出一种多策略改进黏菌算法(enhanced slime mould algorithm with multi-strategy, MESMA)。MESMA 采用的改进策略如下:①采用佳点集初始化种群,提高种群遍历性;②引入广义正态分布搜索策略,以当前的平均位置和最好位置为指导更新黏菌种群位置,提高算法开发能力;③加入自适应比例变异策略,进一步提高算法搜索效率。在 CEC2017 测试集上进行数值实验,通过与其它智能算法及改进的 SMA 进行对比分析,发现 MESMA 具有更好的寻优性能和收敛速度。最后,采用改进算法 MESMA 对 SVM 的两个参数进行优化,建立 MESMA-SVM 分类模型,并在 UCI 数据库中的 4 个数据集上进行对比实验,结果显示 MESMA-SVM 相比于 PSO-SVM、GWO-SVM 和 SMA-SVM 具有更高的分类准确率。

1 SVM 基本理论

支持向量机的主要思想是在特征空间中寻找最

佳分离超平面,使映射到高维特征空间的支持向量到该超平面的距离最大化。给定训练样本集 $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_l, y_l)\}$, $y_i \in \{-1, 1\}$, 设 $\phi(\mathbf{x})$ 为样本空间到高维特征空间的映射函数, $\mathbf{w}^\top \phi(\mathbf{x}) + b = 0$ 为特征空间中的划分超平面,其中 \mathbf{w} 为超平面法向量, b 为位移项。寻找最优分类超平面属于一个优化问题,具体如下:

$$\begin{cases} \min \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^l \xi_i \\ \text{s. t. } y_i(\mathbf{w}^\top \phi(\mathbf{x}_i) + b) \geq 1 - \xi_i, i = 1, 2, \dots, l \end{cases} \quad (1)$$

式中: $\xi_i \geq 0$ 为松弛变量; C 为惩罚因子。

使用拉格朗日乘子法,可得式(1)的对偶优化问题为:

$$\begin{cases} \min \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^l \alpha_i \\ \text{s. t. } \sum_{i=1}^l \alpha_i y_i = 0, 0 \leq \alpha_i \leq C, i = 1, 2, \dots, l \end{cases} \quad (2)$$

式中: $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle = \phi(\mathbf{x}_i)^\top \phi(\mathbf{x}_j)$ 为核函数; α_i 为拉格朗日乘子。

对式(2)进行求解,可确定最优分离超平面,从而可得决策函数:

$$f(\mathbf{x}) = \text{sgn} \left(\sum_{i=1}^l \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \right) \quad (3)$$

式中: $\text{sgn}(\cdot)$ 为符号函数。

径向基核函数(RBF)可逼近任何非线性函数,径向基核函数为:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma^2) \quad (4)$$

式中: $\sigma > 0$ 为核参数。

由于 RBF 参数 σ 和惩罚因子 C 对 SVM 模型的分类性能影响较大,本文通过改进的黏菌算法对 SVM 的参数进行优化。

2 改进的黏菌算法

2.1 基本的黏菌算法

2.1.1 接近食物

黏菌根据空气中的气味接近食物,其收缩模式的模拟公式为:

$$\mathbf{X}(t+1) = \begin{cases} \mathbf{X}_b(t) + \mathbf{v}_b \cdot (\mathbf{W} \cdot \\ \mathbf{X}_A(t) - \mathbf{X}_B(t)), & r < p \\ \mathbf{v}_c \cdot \mathbf{X}(t), & r \geq p \end{cases} \quad (5)$$

式中: t 为当前迭代次数; \mathbf{X} 为黏菌的位置; \mathbf{X}_b 为当前发现的气味浓度最高的个体位置,即最佳个体位置; \mathbf{X}_A 和 \mathbf{X}_B 为在黏菌种群中随机选取的两个个体;

v_b 为 $[-a, a]$ 之间的参数; v_c 随着迭代次数从 1 线性递减至 0; W 为黏菌的权重; r 为 $[0, 1]$ 区间的随机数。

p 的计算公式为:

$$p = \tanh |S(i) - D_F| \quad (6)$$

式中: $S(i)$ 为黏菌个体 X 的适应度值, $i = 1, 2, \dots, N$; N 为种群规模; D_F 为当前得到的黏菌个体的最佳适应度值。

v_b 的计算公式为:

$$v_b = [-a, a] \quad (7)$$

$$a = \operatorname{arctanh}\left(-\frac{t}{T} + 1\right) \quad (8)$$

式中: T 为最大迭代次数。

W 的计算公式为:

$$W(S_1(i)) = \begin{cases} 1 + r \cdot \log\left(\frac{b_F - S(i)}{b_F - w_F} + 1\right), & C_{on} \\ 1 - r \cdot \log\left(\frac{b_F - S(i)}{b_F - w_F} + 1\right), & \text{其他} \end{cases} \quad (9)$$

$$S_1 = \operatorname{sort}(S) \quad (10)$$

式中: b_F 和 w_F 分别为当前迭代下获得的最佳和最差适应度值; C_{on} 为 $S(i)$ 在种群中排序前一半的部分; S_1 为根据适应度值排序的序列(在最小值问题中为升序)。

2.1.2 包裹食物

基于黏菌包裹食物时的行为, 黏菌的位置更新公式为:

$$X^* = \begin{cases} r' \cdot (u_b - l_b) + l_b, & r' < z \\ X_b(t) + v_b \cdot (W \cdot X_A(t) - X_b(t)), & r < p \\ v_c \cdot X(t), & r \geq p \end{cases} \quad (11)$$

式中: r' 为 $[0, 1]$ 区间的随机数; u_b 和 l_b 表示搜索空间的上界和下界; $z = 0.03^{[10]}$ 。

2.1.3 振荡过程

v_b 在 $[-a, a]$ 区间随机振荡, 并随着迭代次数的增加逐渐趋于零, 其振荡过程模拟了黏菌决定接近食物还是寻找其它食物源。 v_c 在 $[-1, 1]$ 区间振荡, 最终趋于零。 v_b 和 v_c 之间的协同交互模拟了黏菌选择食物的过程。

2.2 多策略改进黏菌算法

2.2.1 佳点集初始化黏菌种群

对于群智能算法, 初始种群的分布多样性很大程度上影响着算法的搜索速度和优化能力, 黏菌算法采取的是随机初始化种群, 其产生的初始种群在解空间的均匀遍历性得不到保证, 会存在个体聚集问

题。文献[14]指出, 采用佳点集生成的点集比随机方式生成的点集误差小了平方倍。因此, 本文采用数论中的佳点集对种群进行初始化, 其原理和结构如下。

设 V_d 是 d 维空间中的单位立方体, 若 $r \in V_d$, 形式(12)的数集集合可称为佳点集。

$$P_n(k) = \{(\{r_1^{(n)} \cdot k\}, \{r_2^{(n)} \cdot k\}, \dots, \{r_d^{(n)} \cdot k\}), 1 \leq k \leq n\} \quad (12)$$

其中, 偏差 $\phi(n)$ 满足 $\phi(n) = C(r, \epsilon)n^{-1+\epsilon}$, $C(r, \epsilon)n^{-1+\epsilon}$ 为只和 r 、 ϵ (ϵ 为任意正数) 有关的常数, r 为佳点, $\{r_d^{(n)} \cdot k\}$ 表示取小数部分, n 为佳点个数, 一般取 $r = \{2\cos(2\pi k/p)\}, 1 \leq k \leq d\}$, p 为满足 $d \leq (p-3)/2$ 的最小素数。

通过式(12)生成佳点集后, 将其映射到搜索空间生成初始种群。在 $[0, 1] \times [0, 1]$ 的空间中采用随机法和佳点集法生成两个包含 100 个点的点集, 如图 1 和图 2 所示, 可以看出, 采用佳点集法生成的点集分布更均匀, 可以有效避免无效个体的产生, 且其高维空间也不失一般性。

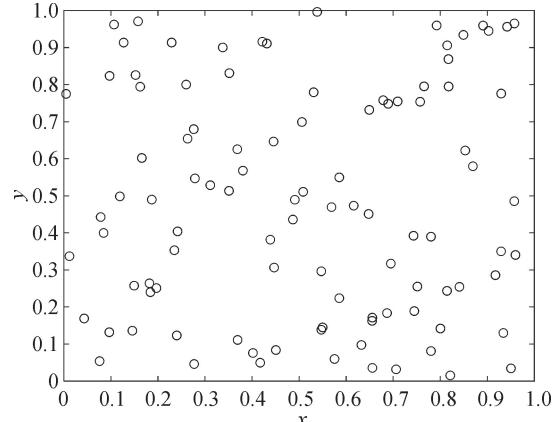


图 1 随机法生成的点集
Fig. 1 Point set generated by random method

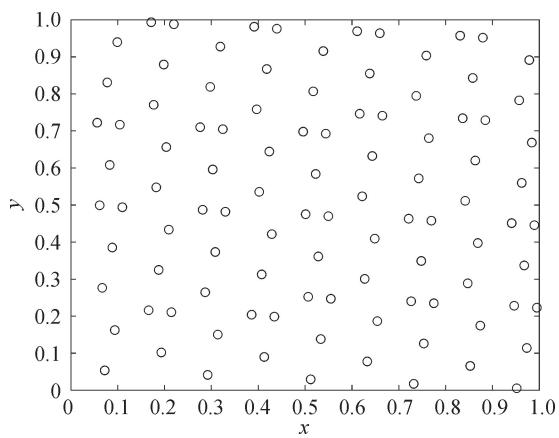


图 2 佳点集法生成的点集
Fig. 2 Point set generated by good point set method

2.2.2 广义正态分布搜索策略

随着迭代次数的增加, p 值逐渐减小, 基本黏菌算法执行 $\mathbf{X}^* = \mathbf{X}_b(t) + v_b \cdot (\mathbf{W} \cdot \mathbf{X}_A(t) - \mathbf{X}_B(t))$, $r < p$ 的可能性变小, 因此在基本 SMA 位置更新之前, 采用广义正态分布优化算法^[15] (generalized normal distribution optimization, GNDO) 中的局部开发策略可以增强 SMA 算法的开发能力。广义正态分布优化算法中的局部开发策略以当前的平均位置和最优位置为指导, 其位置更新公式为:

$$\mathbf{X}_i(t+1) = \boldsymbol{\mu}_i + \eta \times \boldsymbol{\delta}_i \quad (13)$$

式中: $i = 1, 2, 3, \dots, N$; $\boldsymbol{\mu}_i$ 为广义平均位置; $\boldsymbol{\delta}_i$ 为广义标准差; η 为参数, 其具体计算公式为:

$$\boldsymbol{\mu}_i = \frac{1}{3}(\mathbf{X}_i(t) + \mathbf{X}_b(t) + \mathbf{M}) \quad (14)$$

$$\boldsymbol{\delta}_i = \sqrt{\frac{1}{3}[(\mathbf{X}_i(t) - \boldsymbol{\mu}_i)^2 + (\mathbf{X}_b(t) - \boldsymbol{\mu}_i)^2 + (\mathbf{M} - \boldsymbol{\mu}_i)^2]} \quad (15)$$

$$\eta = \begin{cases} \sqrt{-\ln(\lambda_1)} \times \cos(2\pi\lambda_2), & a \leq b \\ \sqrt{-\ln(\lambda_1)} \times \cos(2\pi\lambda_2 + \pi), & \text{其他} \end{cases} \quad (16)$$

$$\mathbf{M} = \frac{\sum_{i=1}^N \mathbf{X}_i(t)}{N} \quad (17)$$

式中: $a, b, \lambda_1, \lambda_2$ 均为 $[0, 1]$ 区间的随机数; \mathbf{M} 为当前种群的平均位置; N 为种群规模。

图 3 显示了由式(16)生成的随机序列, 可以看到参数 η 大多处于 -1 到 1 之间, 可通过增加黏菌个体搜索的方向来提高算法的搜索能力。

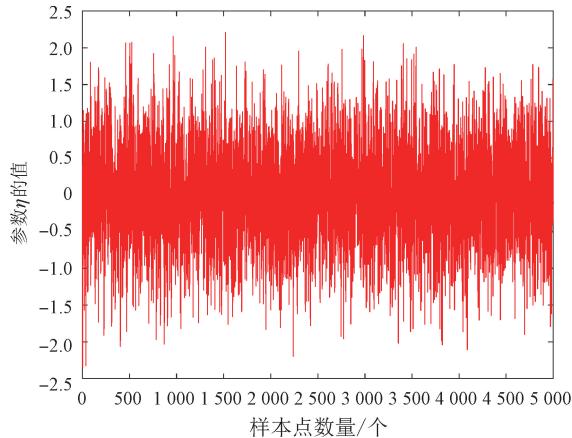


图 3 式(16)生成的随机序列

Fig. 3 Random sequence generated by Equation (16)

2.2.3 自适应比例变异策略

为进一步提高 SMA 的搜索效率, 在基本 SMA 位置更新之后加入自适应比例变异策略^[16], 即在进入下一次迭代之前, 根据适应度值对个体进行排序(极小化问题为升序, 极大化问题为降序), 保留个体

较好的前 $1/2$ 种群, 而对后 $1/2$ 种群的个体进行变异操作, 变异个体的比例随迭代次数的增加呈非线性下降。执行自适应比例变异策略的位置更新如下:

$$X_{2_no} = \text{floor}((N/2) \times t) \quad (18)$$

$$X_{1_no} = N - X_{2_no} \quad (19)$$

$$t = (1 - \frac{t}{T})^{\frac{t}{T}} \quad (20)$$

$$X_1 = X(1:X_{1_no}, :) \quad (21)$$

$$X_2 = X(X_{1_no} + 1:X_{end}, :) \quad (22)$$

$$X_2 = X_2 + \text{rand}(1, \text{dim}) \times (X(1, :) - X_2) \quad (23)$$

$$X = [X_1, X_2] \quad (24)$$

式中: $\text{floor}(\cdot)$ 表示向下取整数; X_{1_no} 和 X_{2_no} 分别为保留个体和变异个体的种群规模; X_1 和 X_2 分别为保留个体种群和变异个体种群; $X(1, :)$ 表示种群排序后的第一个个体位置, 即当前迭代下的最佳个体位置; X_{end} 为种群中最后一个个体。

自适应比例变异策略可以快速淘汰适应度值较差的个体, 早期变异比例较大可以进一步提高算法的全局搜索能力, 更快地找到全局最优区域, 而后期变异比例较小可以保留更占优势的个体, 从而有效提高算法的搜索效率。

2.3 MESMA 算法步骤

根据以上三种策略提出 MESMA 算法的具体执行步骤:

步骤 1: 初始化参数, 包括种群规模 N , 问题维数 D , 最大迭代次数 T , 参数 z , 搜索空间的上界 u_b 、下界 l_b ;

步骤 2: 采用佳点集初始化种群位置。利用式(12)生成佳点集, 然后在搜索空间生成初始黏菌种群;

步骤 3: 计算每个黏菌个体的适应度值, 并对适应度值进行排序, 更新当前迭代次数下最好的适应度值 b_F 、最坏的适应度值 w_F 以及当前的最佳个体位置 \mathbf{X}_b ;

步骤 4: 根据式(13)~(17)进行广义正态分布搜索, 计算每个黏菌个体的适应度值并更新当前最佳个体位置;

步骤 5: 利用式(6)~(9)计算参数 p 、 v_b 和权重 \mathbf{W} , 根据式(11)更新黏菌位置;

步骤 6: 计算适应度值并对个体进行排序, 然后采用式(18)~(24)执行自适应比例变异操作;

步骤 7: 判断是否达到最大迭代次数, 若是则结束循环, 输出最优个体位置, 否则返回步骤 3。

3 MESMA 性能测试

在本节实验中,选取 CEC2017 基准测试集来测试 MESMA 算法的综合优化性能。

3.1 测试函数的选择及参数设置

CEC2017 基准测试集共包含 30 个函数,均为最小化问题,搜索范围均为 $[-100, 100]^D$, D 为问题维数。本文选取 F1、F3(单峰函数),F5、F9(多峰函数),F11、F15、F18 与 F19(混合型函数),F21、F30(复合型函数)这 10 个函数进行数值实验。所选函数涉及 CEC2017 基准测试集中四种不同的函数类型,因此可以很好地检验算法在解决不同数值优化问题时的能力。为了进行对比实验,所有算法

均设置相同的参数:种群大小 N 为 30、问题维数 D 为 30、算法终止准则为最大迭代次数 $T = 1000$ 。

3.2 MESMA 测试与结果分析

将 MESMA 算法同粒子群优化(PSO)^[17]、灰狼优化(GWO)^[18]、正余弦算法(SCA)^[19]、SMA^[10]、平衡黏菌算法(ESMA)^[20]、基于反向学习的黏菌算法(OSMA)^[21]进行对比。其中 PSO 的惯性权重分别为 $w_{\max} = 0.95$ 、 $w_{\min} = 0.4$, 学习因子分别为 $c_1 = 2.0$ 、 $c_2 = 2.0$; GWO 中 a 从 2 到 0 线性递减; SCA 中 $a = 2$; SMA、ESMA、OSMA 和所提 MESMA 中的 z 均取 0.03。为了避免随机因子和初始种群的随机性产生的偶然误差,令每种算法独立寻优 30 次,记录误差平均值(Mean)与标准差(Std),具体结果如表 1 所示。

表 1 MESMA 和其它智能算法在 CEC2017 部分测试函数上的实验结果

Tab. 1 Experimental results by MESMA and other intelligent algorithms on CEC2017 partial test functions

函数	指标	GWO	PSO	SCA	SMA	ESMA	OSMA	MESMA
F1	Mean	2.14E+09	3.78E+09	1.87E+10	1.30E+04	1.07E+04	1.71E+04	9.11E+03
	Std	1.49E+09	2.03E+10	3.07E+09	7.50E+03	6.78E+03	6.80E+03	6.50E+03
F3	Mean	5.05E+04	3.20E+04	6.54E+04	5.19E+03	1.41E+04	9.98E+03	4.55E+03
	Std	1.23E+04	6.57E+04	1.16E+04	4.59E+03	7.34E+03	6.22E+03	2.19E+03
F5	Mean	1.13E+02	1.64E+02	3.09E+02	1.18E+02	1.09E+02	1.33E+02	7.83E+01
	Std	3.20E+01	3.17E+01	2.66E+01	2.79E+01	3.07E+01	3.47E+01	1.99E+01
F9	Mean	1.18E+03	3.78E+03	6.18E+03	3.86E+03	2.76E+03	3.70E+03	1.12E+02
	Std	6.19E+02	4.61E+03	1.29E+03	1.74E+03	1.76E+03	1.54E+03	1.41E+02
F11	Mean	1.16E+03	1.31E+02	2.14E+03	1.81E+02	1.50E+02	1.36E+02	7.84E+01
	Std	9.63E+02	4.54E+01	8.86E+02	4.58E+01	4.90E+01	5.53E+01	3.08E+01
F15	Mean	2.61E+06	1.77E+08	3.42E+07	2.21E+04	1.59E+04	1.49E+04	3.50E+03
	Std	9.87E+06	6.74E+08	2.81E+07	1.49E+04	1.40E+04	1.41E+04	3.04E+03
F18	Mean	1.29E+06	4.78E+07	9.38E+06	2.42E+06	1.96E+06	8.45E+05	2.79E+05
	Std	2.45E+06	1.80E+08	6.28E+06	2.07E+06	1.96E+06	8.07E+05	2.05E+05
F19	Mean	1.18E+06	6.14E+08	6.74E+07	3.44E+04	2.51E+04	1.43E+04	3.67E+03
	Std	1.97E+06	2.23E+09	2.87E+07	2.02E+04	2.33E+04	1.69E+04	4.21E+03
F21	Mean	3.07E+02	4.06E+02	4.88E+02	3.20E+02	3.15E+02	3.40E+02	2.71E+02
	Std	3.58E+01	1.39E+02	2.37E+01	3.50E+01	3.16E+01	2.88E+01	1.88E+01
F30	Mean	1.10E+07	8.39E+08	1.53E+08	4.49E+04	1.51E+04	2.61E+04	9.04E+03
	Std	1.27E+07	1.79E+09	7.34E+07	2.86E+04	7.24E+03	1.77E+04	3.89E+03

注:粗体数据表示同一函数下相关指标的最优值。

由表 1 可知, MESMA 算法在求解不同类型的函数时具有明显优势, 其在 10 个函数上的均值、标准差都小于其它 6 种算法, 特别是与 GWO 和 SCA 算法相比, MESMA 的求解精度更高, 说明引入策

略能够有效提高 SMA 的收敛速度和稳定性。

为了直观比较 MESMA 与其它 6 种智能算法的优化性能, 绘制 MESMA 和其它 6 种算法的收敛曲线如图 4 所示。

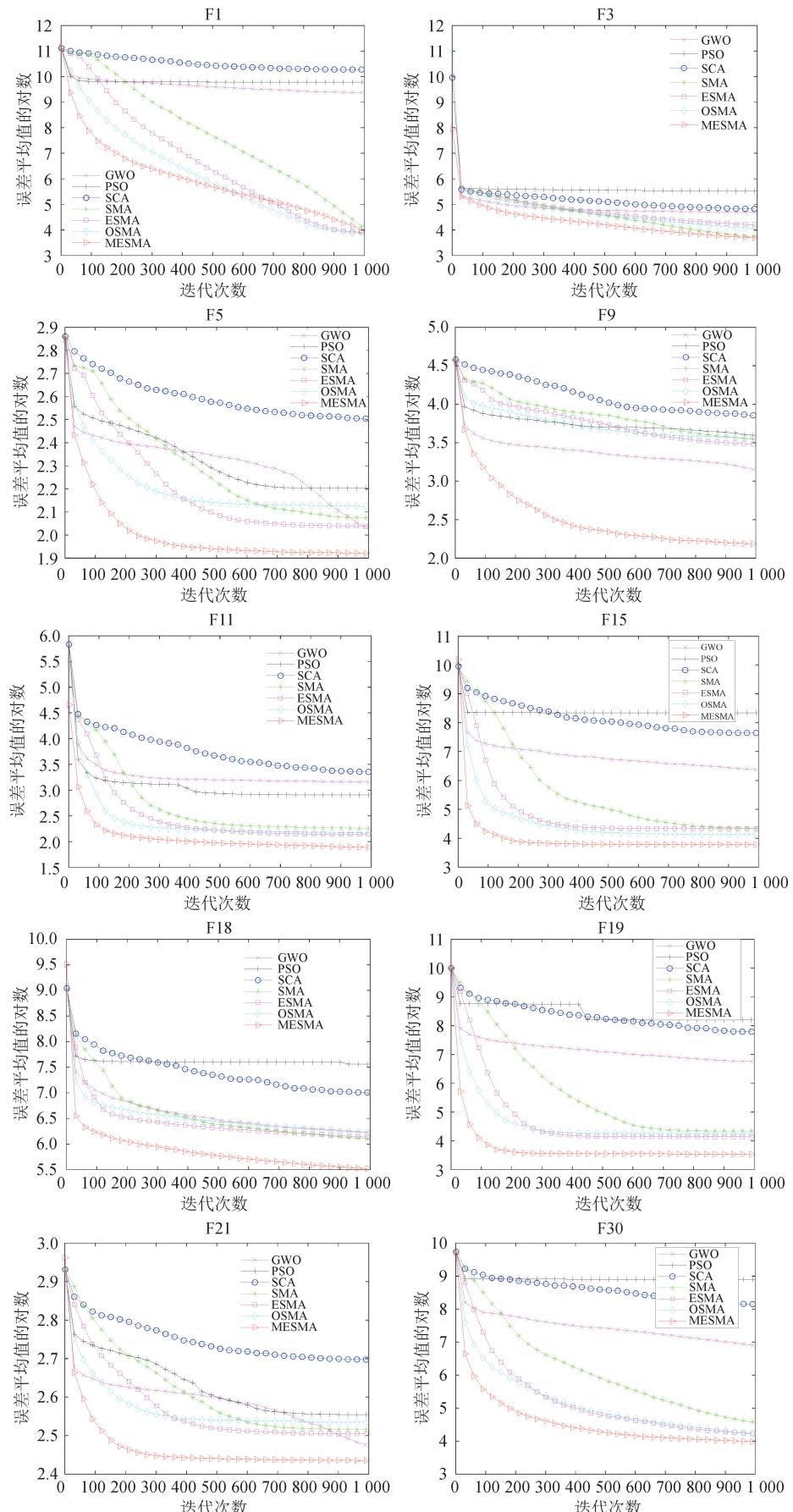


图 4 CEC2017 部分测试函数上的收敛曲线

Fig. 4 Convergence curves on some test functions of CEC2017

从图 4 可以清楚地看到,10 种不同类型函数下,MESMA 的收敛曲线均位于其他曲线的最下方,说明它可以在较少的迭代次数下达到更高的寻优精度,在迭代 100 次时 MESMA 就明显优于其它算法。对于函数 F18, MESMA 在迭代到 1 000 次时仍有下降的趋势,说明该算法可有效避免陷入局部最优。对于函数 F9、F18、F19 和 F21, MESMA 在迭代到 200 次时就达到了较优的收敛精度,且在迭代到 100 次时收敛精度就开始优于其它 6 种算法,说明 MESMA 算法在求解复杂问题时拥有更高的收敛精度。综上所述,MESMA 在 CEC2017 测试集上的寻优精度和收敛速度均优于其它 6 种智能算法。

3.3 策略有效性分析

为了测试三种策略对基本 SMA 优化性能的贡献程度和三种策略结合使用的必要性,将仅引入佳点集初始化种群的 SMA 命名为 GPSSMA,将仅引入广义正态分布搜索策略的 SMA 命名为 GNDSMA,将仅引入自适应比例变异策略的 SMA 命名为 APMSMA,然后同 SMA 和 MESMA 一起,在 CEC2017 基准测试集的 10 个函数下进行数值实验。为了避免初始种群的随机性和随机因子产生的偶然误差,令每个算法在各函数上独立运行 30 次,记录其误差平均值(Mean)和标准差(Std),并对结果进行排名(Rank),实验结果如表 2 所示。

表 2 不同改进策略在 CEC2017 部分测试函数上的实验结果

Tab. 2 Experimental results of different improvement strategies on CEC2017 partial test functions

函数	指标	GPSSMA	GNDSMA	APMSMA	SMA	MESMA
F1	Mean	1.3784E+04	9.9773E+04	1.2875E+04	1.4588E+04	5.8063E+03
	Std	7.5997E+03	5.1720E+04	7.8497E+03	9.5324E+03	4.3015E+03
	Rank	3	5	2	4	1
F3	Mean	7.2630E+03	1.1253E+04	4.4603E+03	6.0223E+03	5.7621E+03
	Std	5.0150E+03	3.9793E+03	2.8483E+03	3.8314E+03	3.1176E+03
	Rank	4	5	1	3	2
F5	Mean	1.1853E+02	9.4740E+01	1.1493E+02	1.2508E+02	7.5918E+01
	Std	3.7214E+01	1.9007E+01	3.2583E+01	3.9608E+01	1.6952E+01
	Rank	4	2	3	5	1
F9	Mean	2.6899E+03	3.5524E+02	2.1944E+03	3.4294E+03	1.3330E+02
	Std	1.2445E+03	2.5501E+02	9.5095E+02	1.6067E+03	1.7302E+02
	Rank	4	2	3	5	1
F11	Mean	1.8022E+02	7.9557E+01	1.5314E+02	1.8450E+02	6.5493E+01
	Std	6.1148E+01	2.7030E+01	4.3032E+01	6.6154E+01	2.4956E+01
	Rank	4	2	3	5	1
F15	Mean	2.3883E+04	5.7834E+03	1.7669E+04	2.4477E+04	3.2616E+03
	Std	1.5124E+04	7.1968E+03	1.5604E+04	1.4147E+04	2.3948E+03
	Rank	4	2	3	5	1
F18	Mean	1.7093E+06	2.9057E+05	1.4907E+06	1.4873E+06	1.7351E+05
	Std	1.9777E+06	2.4264E+05	1.3275E+06	1.4514E+06	8.9594E+04
	Rank	5	2	4	3	1
F19	Mean	2.1406E+04	4.9452E+03	1.9762E+04	2.4422E+04	4.2843E+03
	Std	2.1064E+04	1.0403E+04	2.1536E+04	2.2701E+04	2.2047E+03
	Rank	4	2	3	5	1

表 2(续)

函数	指标	GPSSMA	GNDSMA	APMSMA	SMA	MESMA
F21	Mean	3.1563E+02	2.9065E+02	3.1201E+02	3.1943E+02	2.7210E+02
	Std	2.9382E+01	1.9114E+01	3.4344E+01	3.8515E+01	1.3875E+01
	Rank	4	2	3	5	1
F30	Mean	5.4332E+04	2.5458E+04	6.3761E+04	4.6678E+04	1.0520E+04
	Std	4.3163E+04	1.0452E+04	7.0144E+04	2.5645E+04	3.9272E+03
	Rank	4	2	5	3	1

注:粗体数据表示同一函数下均值、标准差的最优值。

从表 2 可以看出, GPSSMA、GNDSMA、APMSMA 在多数函数上的误差平均值和标准差都优于 SMA, 而结合三种策略的 MESMA 相比这三种仅引入单个策略的黏菌算法寻优结果更好、在几乎所有函数中排名第一, 说明三种策略的结合使用是必要的, 可有效提高原始 SMA 的收敛速度、收敛精度和稳定性。

4 基于 MESMA 的 SVM 参数优化

4.1 采用 MESMA 对 SVM 参数进行优化

SVM 的性能与惩罚因子 C 和核参数 σ 密切相关, 结合上节测试结果, 利用改进算法 MESMA 对 SVM 模型的两个参数进行寻优, 即构建 MESMA-SVM 分类模型, 以保证 SVM 分类器具有较高的分类准确率。

采用 MESMA-SVM 模型的主要目的是提高分类准确率, 因此选取分类准确率来构建 MESMA 算法的适应度函数:

$$f = \text{avg}(A_{\text{cc}}) = \frac{1}{k} \sum_{i=1}^k A_{\text{cc},i} \quad (25)$$

式中: $A_{\text{cc},i}$ 为第 i 折交叉验证的准确率; f 为 k 折交叉验证计算出的平均分类准确率。

采用 MESMA 优化 SVM 参数的步骤如下:

步骤 1: 输入数据集, 对原始数据进行归一化处理;

步骤 2: 初始化参数, 将 SVM 的待优化参数 C 和 σ 作为黏菌的位置参数, 对黏菌算法的输入参数进行初始化, 包括种群规模大小 N 、搜索空间上界 u_b 和下界 l_b (即惩罚因子 C 和核参数 σ 的寻优范围)、最大迭代次数 T 和问题维数 D ;

步骤 3: 以式(25)为优化目标, 采用 MESMA 对参数进行自适应寻优;

步骤 4: 判断是否达到最大迭代次数, 若达到则终止算法, 输出最优黏菌位置 \mathbf{X}_b , 即最佳参数组合

(C, σ) , 否则返回步骤 3。

4.2 SVM 评价指标

采用分类准确率(A_{cc})、查准率(P_{re})、查全率(R_{ec})和 F_1 值对 MESMA-SVM 模型的分类性能进行评估。四个指标的值越大表示模型的分类效果越好。其表达式分别为:

$$A_{\text{cc}} = \frac{TP + TN}{TP + FP + FN + TN} \times 100\% \quad (26)$$

$$P_{\text{re}} = \frac{TP}{TP + FP} \quad (27)$$

$$R_{\text{ec}} = \frac{TP}{TP + FN} \quad (28)$$

$$F_1 = \frac{2 \times P_{\text{re}} \times R_{\text{ec}}}{P_{\text{re}} + R_{\text{ec}}} \quad (29)$$

式中: TP 为真正例; FP 为假正例; TN 为真反例; FN 为假反例。

4.3 实验与结果分析

为验证 MESMA-SVM 模型的分类性能, 将其与 PSO 优化的 SVM(PSO-SVM)、GWO 优化的 SVM(GWO-SVM) 和 SMA 优化的 SVM(SMA-SVM) 三种模型进行比较。选取加州大学欧文分校创建的 UCI 机器学习数据库(<http://archive.ics.uci.edu/>)中的 4 个标准数据集进行对比实验, 各数据集的描述如表 3 所示, 由于每个数据集的特征值的数量级不同, 在构建 SVM 模型之前, 将数据归一化到 $[0, 1]$ 区间。实验采用林智仁教授团队开发的 LIBSVM 工具箱在 MATLAB 2019a 中实现, 为保证结果的有效性, 采用五折交叉验证法。PSO 和 GWO 算法的参数设置与 3.2 节相同, 同时为了体现实验的公平性, 所有算法的共有参数均设置为相同值: 种群规模为 20, 迭代次数为 100, 待优化参数 C 和 σ 的搜索范围为 $[0.01, 1000]$ 。

表 3 数据集描述
Tab. 3 Description of datasets

数据集	样本数	特征数	类别数
Ionosphere	351	34	2
Seeds	210	7	3
Vehicle	846	18	4
Aggregation	788	2	7

表 4 展示了 PSO-SVM、GWO-SVM、SMA-SVM 和 MESMA-SVM 四种模型分别在 Ionosphere、Seeds、Vehicle 和 Aggregation 四个数据集上的评价指标值。

在数据集 Ionosphere 和 Aggregation 上, MESMA-SVM 模型的 A_{cc} 均超过了 99%, 其 P_{re} 、 R_{cc} 与 F_1 值也都取得了最优结果。在数据集 Seeds 上, MESMA-SVM 模型的 A_{cc} 、 P_{re} 、 R_{cc} 与 F_1 值分别为 93.650 8%、0.914 1、0.904 8 和 0.909 4, 较 PSO-SVM、GWO-SVM 和 SMA-

SVM 模型有显著提高。在数据集 Vehicle 上, MESMA-SVM 模型的四个指标值也均为最优。实验结果表明, 本文采用 MESMA 算法对 SVM 参数进行寻优所建立的 MESMA-SVM 模型是一个优异的分类模型, 可以有效提高 SVM 的分类准确率。究其原因, SMA 可在实现优越的开发能力的同时保证其探索能力, 从而在开发和探索之间保持良好的平衡^[10]。MESMA 在原始 SMA 基础上加入了佳点集初始化种群、广义正态分布搜索和自适应比例变异策略, 进一步提高了 SMA 算法的种群均匀性、局部开发能力和收敛速度, 这使得 MESMA 能够有效避免传统优化算法易陷入局部最优的问题, 使其相较于 PSO、GWO 和 SMA 拥有更好的优化性能, 因此, 采用 MESMA 优化的 SVM 模型具有更高的分类准确率和指标值。

表 4 4 种模型在 4 个数据集上的对比结果
Tab. 4 Comparison results by 4 models on each of 4 datasets

数据集	模型	A_{cc} /%	P_{re}	R_{cc}	F_1
Ionosphere	PSO-SVM	65.094 3	0.823 8	0.513 2	0.632 4
	GWO-SVM	97.169 8	0.972 0	0.966 3	0.969 2
	SMA-SVM	97.169 8	0.972 0	0.966 3	0.969 2
	MESMA-SVM	99.056 6	0.985 7	0.973 7	0.979 7
Seeds	PSO-SVM	39.682 5	0.785 3	0.396 8	0.527 2
	GWO-SVM	84.127 0	0.878 0	0.841 3	0.859 3
	SMA-SVM	84.127 0	0.878 0	0.841 3	0.859 3
	MESMA-SVM	93.650 8	0.914 1	0.904 8	0.909 4
Vehicle	PSO-SVM	36.177 5	0.338 2	0.264 9	0.297 1
	GWO-SVM	87.372 0	0.876 7	0.881 4	0.879 0
	SMA-SVM	87.030 7	0.873 5	0.877 5	0.875 5
	MESMA-SVM	91.601 1	0.913 4	0.886 3	0.899 6
Aggregation	PSO-SVM	70.042 2	0.917 9	0.672 4	0.776 2
	GWO-SVM	94.936 7	0.968 2	0.915 7	0.941 2
	SMA-SVM	98.312 2	0.981 8	0.973 7	0.977 7
	MESMA-SVM	99.367 1	0.991 3	0.976 5	0.983 8

注: 粗体数据表示不同数据集下各评价指标的最优值。

5 结语

本文提出了一种基于 MESMA 的支持向量机参数优化方法, 在一定程度上解决了支持向量机惩罚因子和核参数的优化选择问题。在 MESMA 中, 采用佳点集初始化种群使黏菌个体的空间分布更均匀, 引入广义正态分布搜索策略增强算法的开发能

力, 采用自适应比例变异策略进一步提高了算法的搜索效率。CEC2017 基准测试集上的数值实验结果表明, MESMA 的寻优精度和收敛速度均优于其它 6 种智能算法。利用 MESMA 对 SVM 的参数进行寻优, 建立 MESMA-SVM 分类模型, 并在 4 个标准数据集上进行对比实验, 结果显示 MESMA-SVM 模型相比其它模型具有更优的分类性能以及

更高的分类准确率。

下一步,将继续探索如何将其它良好策略与SMA有效结合来提高SMA的寻优性能,并考虑将MESMA-SVM模型用于解决生活中的实际问题。

参考文献:

- [1] CORTES C, VAPNIK V. Support-vector networks [J]. *Machine Learning*, 1995, 20: 273-297.
- [2] WANG Zheya, LI Gaosong, YAO Ligang, et al. Intelligent fault detection scheme for constant-speed wind turbines based on improved multiscale fuzzy entropy and adaptive chaotic Aquila optimization-based support vector machine[J]. *ISA Transactions*, 2023, 138: 582-602.
- [3] BEHERA M P, SARANGI A, MISHRA D, et al. A hybrid machine learning algorithm for heart and liver disease prediction using modified particle swarm optimization with support vector machine[J]. *Procedia Computer Science*, 2023, 218: 818-827.
- [4] ZHENG Nan, ZHANG Guoying, ZHANG Yang, et al. Brain tumor diagnosis based on Zernike moments and support vector machine optimized by chaotic arithmetic optimization algorithm[J]. *Biomedical Signal Processing and Control*, 2023, 82: 104543.
- [5] WANG Yanfeng, LIU Qing, YANG Yuli, et al. Prognostic staging of esophageal cancer based on prognosis index and cuckoo search algorithm-support vector machine[J]. *Biomedical Signal Processing and Control*, 2023, 79: 104207.
- [6] 李素,袁志高,王聪,等.群智能算法优化支持向量机参数综述[J].*智能系统学报*,2018,13(1): 70-84.
LI Su, YUAN Zhigao, WANG Cong, et al. Optimization of support vector machine parameters based on group intelligence algorithm[J]. *CAAI Transactions on Intelligent Systems*, 2018, 13(1): 70-84.
- [7] SHEN Liming, CHEN Huiling, YU Zhe, et al. Evolving support vector machines using fruit fly optimization for medical data classification[J]. *Knowledge-Based Systems*, 2016, 96: 61-75.
- [8] SEYEDMOHAMMADI J, ZEINADINI A, NAVIDI M, et al. A new robust hybrid model based on support vector machine and firefly meta-heuristic algorithm to predict pistachio yields and select effective soil variables[J]. *Ecological Informatics*, 2023, 74, 102002.
- [9] THARWAT A, HASSANIEN A E, ELNAGHI B E. A BA-based algorithm for parameter optimization of support vector machine[J]. *Pattern Recognition Letters*, 2017, 93: 13-22.
- [10] LI Shimin, CHEN Huiling, WANG Mingjing, et al. Slime mould algorithm: a new method for stochastic optimization[J]. *Future Generation Computer Systems*, 2020, 111: 300-323.
- [11] LIU Yun, HEIDARI A A, YE Xiaojia, et al. Boosting slime mould algorithm for parameter identification of photovoltaic models[J]. *Energy*, 2021, 234: 121164.
- [12] BARSHANDEH S, KOULAEZADEH S, MASDARI M, et al. A learning-based metaheuristic administered positioning model for 3D IoT networks[J]. *Applied Soft Computing*, 2023, 136: 110113.
- [13] NAMA S. A novel improved SMA with quasi reflection operator: performance analysis, application to the image segmentation problem of Covid-19 chest X-ray images[J]. *Applied Soft Computing*, 2022, 118: 108483.
- [14] 张铃,张钹.佳点集遗传算法[J].*计算机学报*,2001,24(9): 917-922.
ZHANG Ling, ZHANG Bo. Good point set based genetic algorithm[J]. *Chinese Journal of Computers*, 2001, 24(9): 917-922.
- [15] ZHANG Yiyi, JIN Zhigang, MIRJALILI S. Generalized normal distribution optimization and its applications in parameter extraction of photovoltaic models [J]. *Energy Conversion and Management*, 2020, 224: 113301.
- [16] LIU Jingsen, LI Wuxin, LI Yu. LWMO: an efficient equilibrium optimizer for complex functions and engineering design problems[J]. *Expert Systems with Applications*, 2022, 198: 116828.
- [17] KENNEDY J, EBERHART R. Particle swarm optimization[C]. *Proceedings of ICNN'95-International Conference on Neural Networks*. IEEE, 1995, 4: 1942-1948.
- [18] MIRJALILI S, MIRJALILI S M, LEWIS A. Grey wolf optimizer[J]. *Advances in Engineering Software*, 2014, 69(3): 46-61.
- [19] MIRJALILI S. SCA: a sine cosine algorithm for solving optimization problems[J]. *Knowledge-based Systems*, 2016, 96: 120-133.
- [20] NAIK M K, PANDA R, ABRAHAM A. An entropy minimization based multilevel colour thresholding technique for analysis of breast thermograms using equilibrium slime mould algorithm[J]. *Applied Soft Computing*, 2021, 113: 107955.
- [21] SHARMA A K, SAXENA A, PALWALIA D K. Oppositional slime mould algorithm: development and application for designing demand side management controller[J]. *Expert Systems with Application*, 2023, 214: 119002.