

# 基于社会力模型算法的 SVM 参数优化

张俊杰 阎高伟 李浩志

(太原理工大学信息工程学院 山西 太原 030024)

**摘要** 为了解决支持向量机(SVM)参数优化的问题,提出一种改进的基于社会力模型群智能优化算法 SFSO(Swarm Optimization algorithm based on Social Force Model)的 SVM 参数优化方法。SFSO 通过期望力和排斥力使算法在全局搜索和局部搜索中能够较好的平衡,利用 SFSO 特有的搜索机制对 SVM 的惩罚因子和径向基函数进行优化,提高 SVM 的分类性能。通过对几个 benchmark 函数和常用的 UCI 数据集进行测试表明:改进后的 SFSO 算法不仅对于求解函数优化问题具有较强的鲁棒性和较高的求解精度,而且经改进 SFSO 算法优化后的 SVM 具有更快的收敛速度和更高的分类准确率。

**关键词** 支持向量机 参数优化 SFSO 算法

中图分类号 TP18 文献标识码 A DOI:10.3969/j.issn.1000-386x.2015.09.056

## SVM PARAMETERS OPTIMISATION BASED ON SOCIAL FORCE MODEL ALGORITHM

Zhang Junjie Yan Gaowei Li Haozhi

(College of Information Engineering, Taiyuan University of Technology, Taiyuan 030024, Shanxi, China)

**Abstract** In order to solve the problem of SVM parameters optimisation, we proposed an improved SVM parameters optimisation method, which is based on swarm intelligence optimisation algorithm of social force model (SFSO). SFSO enables the algorithm to be able in well balance in both global and local searches through expectation force and exclusion force. We use special search mechanism of SFSO to optimise the penalty factor of SVM and the radial basis function, thus improve the classification performance of SVM. It is demonstrated through tests on several benchmark functions and common UCI datasets that the improved SFSO algorithm has stronger robustness and higher precision in solving function optimisation problems. Moreover, the SVM optimised by the improved SFSO has faster convergence speed and higher classification accuracy.

**Keywords** Support vector machine (SVM) Parameter optimisation SFSO algorithm

## 0 引言

支持向量机<sup>[1]</sup>SVM 是 Vapnik 等人基于统计学习理论和结构风险最小化原则提出的一种强大的学习机器。相对于传统的学习方法,SVM 能够很好地解决维数灾、小样本以及非线性等难题,目前已被应用于图像分类<sup>[2]</sup>、故障诊断<sup>[3]</sup>、模式识别<sup>[4]</sup>、交通流量预测<sup>[5]</sup>等领域。

SVM 虽然有着完备的理论基础,但在应用中仍然存在问题,比如其参数的选取。实验表明,模型参数对 SVM 的性能有着至关重要的影响。传统的直观经验法或实验法不仅计算量大、效率低,而且选取的往往是局部最优参数。随着智能优化技术不断发展,智能优化算法为 SVM 参数选取提供了一条有效途径<sup>[6,7]</sup>。文献[8,9]提出两种基于遗传算法 GA 的 SVM 参数选择方法,结果表明利用 GA 对 SVM 参数优化不仅缩短了计算时间,还减少了对初值选取的依赖度,但是基于 GA 的方法需要针对不同的优化问题设计不同的交叉、变异算子。文献[10]利用粒子群算法 PSO 对 SVM 参数进行优化,取得了比较好的效果,但是基于 PSO 的方法全局搜索能力差、易陷入局部最优。

受不同物种的启发,研究者们提出许多基于不同搜索机

制的智能优化算法,但在求解不同问题时,这些算法普遍存在一些缺陷。如 GA 与人工鱼群算法 AFSA 在算法后期收敛速度较慢,并且 AFSA 通常仅仅可以获取较为满意解的区域,很难得到精确解;PSO 的单向信息共享虽然加快了收敛速度,但粒子过早的聚集使算法易陷入局部最优,同样人工蜂群算法 ABC 在迭代后期种群多样性缺失,也会遇到陷入局部最优的问题。事实上,目前还没有一种算法可以完全满足人们的需要,因此,改进已有算法或提出基于不同机制的算法用于解决不同优化问题是一项具有重要现实意义的研究课题。

社会力模型<sup>[11]</sup>是 Dirk Helbing 等人提出刻画人群动力学的仿真模型,模拟了人群疏散过程中自组织现象以及拥挤避退等行为。然而,社会力模型忽略了人类社会的协作互助行为。将社会力模型同人类的协作互助行为相结合,本文在基于社会力模型群智能优化算法<sup>[12]</sup>SFSO 的基础上进行了改进。改进后的 SFSO 算法在排斥力的作用下可以有效地避免算法陷入局部最优,单维协作与多维协作相结合,提高了算法的收敛速度和求

收稿日期:2014-03-17。国家自然科学基金项目(60975032);山西省自然科学基金项目(2011011012-2)。张俊杰,硕士生,主研领域:智能控制,优化控制理论及应用,进化计算。阎高伟,教授。李浩志,本科生。

解精度,使算法具有良好的“探索”和“开发”平衡能力。将该算法用于 SVM 参数优化,提高了分类精度,为 SVM 的参数选取提供了一种有效的方法。

## 1 基于社会力模型优化算法

### 1.1 社会力模型

社会力模型作为一种多个自驱动系统的框架,体现了群集智能特性和自组织行为。行人在期望力的作用下向目标运动,当行人间距离较小时,排斥力就会使行人的运动方向发生改变。社会力模型用力的方法描述了行人之间以及行人与建筑之间的非线性作用,用期望力来反映个体对目标的期望作用,用排斥力来解释个体之间保持一定距离的原因,用个体与墙之间的排斥力来反映个体对墙的心理作用。

### 1.2 改进 SFSO 算法

文献[12]提出的 SFSO 算法虽然在二维多模态函数求解的实验中表现出较强的搜索能力,但在实践中仍然存在很多问题,比如对高维函数的求解精度较低、收敛速度慢、容易陷入局部最优。对于某些测试函数实验表明:(1) SFSO 算法在目标选择时采用阈值因子线性调整策略,影响了算法的收敛效率;(2) 行人期望速度的确定不能根据优化问题的解空间大小进行自适应调整;(3) 在算法迭代后期,排斥力已经减至很小时,速度项仍然很大,使行人不能在目标附近求精,影响了最终的求解精度。

针对以上问题,本文对 SFSO 算法进行改进,主要包括:(1) 在目标选择阶段采用概率选择的机制,防止陷入局部最优;(2) 建立期望速度可以根据解空间大小自适应调整的机制;(3) 协作阶段采用单维更新和多维更新,加快了收敛速度并提高了求解精度。算法伪代码如下:

Algorithm 1 改进社会力模型优化算法

Input: 算法参数和求解问题

Output: 优化结果

- 1: Step1: 初始化
- 2: while  $n < t_{max}$  do
- 3: Step2: 选择目标;
- 4: Step3: 社会力驱动;
- 5: Step4: 更新位置;
- 6: Step5: 种群评价;
- 7: Step6: 更新历史记忆;
- 8: Step7: 协作;
- 9:  $n \leftarrow n + 1$
- 10: end while

#### 1.2.1 初始化

初始化主要包括行人的数量  $N$ , 最大迭代次数  $t_{max}$ , 当前位置  $p_\alpha$ , 历史记忆  $h_\alpha$ , 行人  $\alpha$  的初始位置由式(1)随机生成:

$$p_{\alpha,i} = l_i + rand \cdot (u_i - l_i) \quad (1)$$

式中  $p_{\alpha,i}$  为  $p_\alpha$  的第  $i$  维分量,  $i = 1, 2, \dots, D$ ,  $rand$  为  $[0, 1]$  之间的随机数,  $u_i$  和  $l_i$  分别为解空间上下限的第  $i$  维分量, 可行解  $p_\alpha$  的适应度值计算公式为:

$$fit(p_\alpha) = \frac{1}{rank} \quad (2)$$

式中  $rank$  为目标函数值的排列序号。

#### 1.2.2 目标选择阶段

为了防止陷入局部最优,保持目标集的多样性是求解的一

个关键问题。本文采用概率选择机制,行人的历史记忆被选入目标集  $T$  的概率按式(3)计算:

$$prob_\alpha = \frac{0.9(fit(h_\alpha) - fit_{min})}{\max(fit(h_\alpha) - fit_{min})} + 0.1 \quad (3)$$

式中  $fit(h_\alpha)$  为行人  $\alpha$  历史记忆的适应度值,  $fit_{min}$  为当代个体适应度最小值,  $\max(\cdot)$  为取最大值运算。当随机数  $rand \leq prob_\alpha$  时,将行人  $\alpha$  的历史记忆  $h_\alpha$  选入目标集  $T$  中。目标集  $T$  选定后通过计算行人  $\alpha$  与目标集中每个目标的距离,从中选取距离最小的作为  $\alpha$  的目标  $T_\alpha$ 。

#### 1.2.3 社会力驱动

改进的 SFSO 算法在计算社会力之前,采用概率选择的方法将子群体  $G_k$  中的个体分为自由个体和非自由个体。在  $G_k$  中,计算每个行人的当前位置与目标的距离,并求出行人对目标的相对距离  $RD_\alpha$ 。依据每个个体的相对距离最终得到行人成为自由个体的概率:

$$\rho_\alpha = \left(\frac{RD_\alpha}{2}\right)^{3.6} \quad (4)$$

式中  $\rho_\alpha$  为行人成为自由个体的概率,  $RD_\alpha$  为行人  $\alpha$  对目标的相对距离。当随机数  $rand \leq \rho_\alpha$  时,行人  $\alpha$  成为自由个体,否则为非自由个体。

群体被分为两类后,分别执行不同的搜索机制:自由个体会放弃当前位置按式(1)执行随机搜索,提高算法的全局搜索能力;非自由个体则会在社会力的驱动下朝着目标运动。行人  $\alpha$  的社会力主要包括期望力  $f_\alpha^d$  以及行人  $\beta$  对  $\alpha$  的排斥力  $f_{\alpha\beta}^r$ , 定义如式(5):

$$f_\alpha = f_\alpha^d + \sum_{\beta \neq \alpha, \beta \in G_k} f_{\alpha\beta}^r \quad (5)$$

行人  $\alpha$  的期望力  $f_\alpha^d$  由式(6)定义:

$$f_\alpha^d(t) = \frac{ac_\alpha(t) - v_\alpha(t)}{\tau} \quad (6)$$

式中  $ac_\alpha(t)$  和  $v_\alpha(t)$  分别为  $t$  时刻行人  $\alpha$  的加速度和速度,  $\tau$  为行人的适应时间, 本文设为 1。加速度  $ac_\alpha(t)$  由式(7)给出:

$$ac_\alpha(t) = e_\alpha(t)v_\alpha^d(t)(1 + Vfac(u - l)) \quad (7)$$

式中  $e_\alpha(t)$ 、 $v_\alpha^d(t)$  分别为  $t$  时刻行人  $\alpha$  的期望方向和期望速度,  $u, l$  为解空间的上下限,  $Vfac$  为速度因子, 本文设为  $5 \times 10^{-6}$ 。 $e_\alpha(t)$  与  $v_\alpha^d(t)$  由式(8)、式(9)给出:

$$e_\alpha = \frac{p_\alpha - T_\alpha}{|p_\alpha - T_\alpha|} \quad (8)$$

$$v_\alpha^d(t) = \lambda \left( \exp\left(\eta + (\varepsilon - \eta) \frac{d_{\alpha, T_k}}{d_{\xi, T_k}}\right) - 1 \right) / \max(u - l) \quad (9)$$

式中  $\lambda$  为期望因子, 设置为  $6.25 \times 10^{-3}$ ,  $\varepsilon, \eta$  为期望速度范围的控制参数。 $d_{\alpha, T_k}$  为行人  $\alpha$  距目标  $T_k$  的距离,  $d_{\xi, T_k}$  为子群体中个体到目标距离的最大值。

行人  $\alpha$  与  $\beta$  的排斥力由式(10)给出:

$$f_{\alpha\beta}^r(t) = A \cdot \exp\left(\frac{R - nd_{\alpha\beta}}{B}\right) d_{\alpha\beta} \quad (10)$$

式中  $A, B$  为排斥力参数, 设为  $A = 10^{-4}$ ,  $B = 0.1$ ,  $R$  为行人半径,  $d_{\alpha\beta}$  为行人  $\alpha$  和  $\beta$  之间的距离,  $nd_{\alpha\beta} = d_{\alpha\beta} / \max(|d_{\alpha\beta}|)$ 。

半径  $R$  与行人当前位置和历史记忆的标准差  $\delta_c, \delta_h$  有关, 每一代群体的标准差和半径的更新采用加权的方式, 用权重因子  $\mu$  对历史代和当前代的标准差、半径进行权重分配, 标准差更新按下式求得:

$$\delta_c^{(n+1)} = (1 - \mu)\delta_c^n + \mu \cdot std(p^{(n+1)}) \quad (11)$$

$$\delta_h^{(n+1)} = (1 - \mu)\delta_h^n + \mu \cdot std(h^{(n+1)}) \quad (12)$$

式中  $\delta_c^n, \delta_h^n$  为第  $n$  代当前种群位置和历史记忆位置的标准差,  $p^{(n+1)}$  和  $h^{(n+1)}$  分别为第  $(n+1)$  代的行人当前位置和历史记忆,  $std(\cdot)$  为求标准差运算。半径的更新如式(13):

$$R^{n+1} = (1 - \mu)R^n + \mu \cdot RX \quad (13)$$

式中  $R^n$  为第  $n$  代的行人半径,  $RX$  为半径调整因子如式(14):

$$RX = \begin{cases} \min\left(\frac{\delta_h^n}{20}, 0.3\right) \cdot \left(\frac{\delta_h^n}{\delta_c^n}\right) & \frac{\delta_h^n}{\delta_h^0} > 0.5 \\ \max\left(\frac{\delta_h^n}{20}, 5\right) \cdot \left(\frac{\delta_h^n}{\delta_c^n}\right) & \frac{\delta_h^n}{\delta_h^0} \leq 0.5 \quad \delta_h^n > \delta_c^n \\ \min\left(\frac{\delta_h^n}{20}, 1\right) \cdot \left(\frac{\delta_h^n}{\delta_c^n}\right) & \frac{\delta_h^n}{\delta_h^0} \leq 0.5 \quad \delta_h^n \leq \delta_c^n \end{cases} \quad (14)$$

行人  $\alpha$  的速度按式(15)进行更新:

$$v_\alpha(t + \Delta t) = v_\alpha(t) + f_\alpha^d(t)\Delta t + f_{\alpha\beta}^r(t)\Delta t \quad (15)$$

式中,  $v_\alpha(t + \Delta t)$  与  $v_\alpha(t)$  分别为行人在  $t + \Delta t$  和  $t$  时刻的速度,  $f_\alpha^d(t)$  和  $f_{\alpha\beta}^r(t)$  分别为行人在  $t$  时刻所受期望力和排斥力。

行人的位置按式(16)进行更新:

$$p_\alpha(t + \Delta t) = p_\alpha(t) + v_\alpha(t)\Delta t \quad (16)$$

式中  $p_\alpha(t + \Delta t)$  和  $p_\alpha(t)$  分别是行人在  $t + \Delta t$  和  $t$  时刻的位置,  $v_\alpha(t)$  是行人速度,本文中  $t$  为代次  $n, \Delta t$  为 1。

#### 1.2.4 历史记忆更新

经过行人的位置与速度更新之后,对行人的位置进行适应度值计算,通过如下方法选择是否进行个体历史记忆的更新:当满足  $\delta_h^n < \delta_c^n$ , 且  $rand_1 < rand_2$ , 则更新该个体的历史记忆。

#### 1.2.5 协作阶段

协作意味着行人间的信息彼此共享,主要包括两方面信息:当前位置信息和历史记忆信息。为了得到更加优秀的解,若新产生解的适应度值高于历史记忆,则用新解更新历史记忆。

改进的 SFSO 算法采用单维协作与多维协作更新历史记忆,分别定义如式(17)、式(18)所示:

$$h'_{\alpha,i} = \begin{cases} h_{\alpha,i} + 2(h_{\alpha,i} - p_{\beta,i}) \cdot (rand - 0.5) & a < 0.3, b < 0.5 \\ h_{\beta,j} & a < 0.3, b \geq 0.5 \\ h_{\alpha,i} + 2(h_{\alpha,i} - h_{\beta,i}) \cdot (rand - 0.5) & a \geq 0.3 \end{cases} \quad (17)$$

$$h'_\alpha = \begin{cases} h_\alpha + 2(h_\alpha - p_\beta) \cdot (rand - 0.5) & a < b \\ h_\alpha + 2(h_\alpha - h_\beta) \cdot (rand - 0.5) & otherwise \end{cases} \quad (18)$$

式中  $i, j$  代表随机选择的维数,  $h'_{\alpha,i}$  为新生成行人  $\alpha$  历史记忆的第  $i$  维分量,  $p_{\beta,i}$  为行人  $\beta$  的当前位置的第  $i$  维分量,  $h'_\alpha$  表示新生成行人  $\alpha$  的历史记忆,  $rand, a$  和  $b$  表示为  $[0, 1]$  的随机数。如果两个随机数满足  $c < d$  时,则按式(17)进行单维协作,否则按式进行多维协作。

## 2 改进 SFSO 的 SVM 参数选择

### 2.1 SVM 原理

SVM<sup>[13]</sup> 的基本思想是将训练样本数据通过非线性映射函数  $\phi$  映射到特征空间  $H$  中  $S = \{(x_i, y_i) \mid x_i \in H, y_i \in \pm 1\}, i = 1, 2, \dots, l$  为给定训练集,其中  $x_i$  表示输入向量,  $y_i$  表示  $x_i$  对应的输出值。SVM 通过寻找满足分类要求的最优超平面,在保证该超平面正确分类的基础上,使得两类数据集的距离最大。因

此,可以将 SVM 的学习过程转化为如下优化问题:

$$\begin{cases} \min \phi(w) = \frac{1}{2} \langle w \cdot w \rangle + C \sum_{i=1}^l \xi_i \\ \text{s. t. } y_i (\langle w \cdot \phi(x_i) \rangle + b) \geq 1 - \xi_i \quad \xi_i \geq 0 \quad i = 1, 2, \dots, l \end{cases} \quad (19)$$

其中  $C$  为惩罚因子,  $w$  为超平面的法向量,  $\xi_i$  为松弛变量。

引入 Lagrange 函数,将二次规划问题转换为相应的对偶问题,如式(20):

$$\begin{cases} \max L(\alpha) = \sum_{i=1}^k \alpha_i - \frac{1}{2} \sum_{i=1}^k \sum_{j=1}^k \alpha_i \alpha_j y_i y_j K(x_i, x_j) \\ \text{s. t. } \sum_{i=1}^k \alpha_i y_i = 0 \quad \alpha_i \geq 0 \quad i = 1, 2, \dots, k \end{cases} \quad (20)$$

其中,  $K(x_i, x_j)$  为核函数,其值受核函数参数  $\sigma$  影响。

### 2.2 改进 SFSO 算法优化 SVM 参数流程

**Step1** 设置算法参数。种群大小  $N$ , 行人的适应时间  $\tau$ , 最大迭代次数  $t_{\max}$ , 速度因子  $Vfac$ , 期望因子  $\lambda$ , 期望速度控制参数  $\varepsilon, \eta$ , 排斥力参数  $A, B$ , 权重因子  $\mu$ 。

**Step2** 读取样本数据并初始化种群。分别生成  $N$  个行人的当前位置和历史记忆,每个行人由  $(C, \sigma)$  组成。

**Step3** 适应度计算。对初始种群进行 SVM 训练并计算个体的适应度,本文采用 SVM 分类的正确率作为适应度函数。

**Step4** 选择目标。按式(3)形成目标集  $T$  并为每个行人选定自己的目标  $T_\alpha$ 。

**Step5** 计算期望力与排斥力。

**Step6** 行人的速度和位置更新。

**Step7** 更新历史记忆。计算更新后的种群个体适应度值,以概率的方式进行历史记忆更新。

**Step8** 协作。对历史记忆采用单维协作或多维协作,保留协作后适应度较高的历史记忆。

**Step9** 终止条件判断。如果  $n > t_{\max}$ , 则算法停止,输出具有最大适应度个体作为最优参数。否则返回 Step4。

## 3 仿真实验

### 3.1 函数优化实验

为了测试改进后 SFSO 算法的有效性,本文选取 6 个 benchmark 函数<sup>[14]</sup>进行测试,其基本特性如表 1 所示。每种算法对 6 个函数独立运行 30 次,记录每次运行的最优值,最终得到平均值(Mean)、标准偏差(Std)、最优值(Best)如表 2 所示。

表 1 Benchmark 函数

Fun	Name	Dim	Low	Up
F1	Shubert	2	-10	10
F2	Griewangk	2	-600	600
F3	Foxholes	2	-65.536	65.536
F4	Weierstrass	30	-0.5	0.5
F5	Quartic	30	-1.28	1.28
F6	Ackley	30	-32	32

从表 2 可以看出不论是低维还是高维函数,改进后的 SFSO 算法都表现出较好的求解效果,精度有了明显的提高。

表 2 SFSO 和改进 SFSO 对函数 F1 - F6 实验结果对比

Fun	算法	Mean	Best	Std
F1	SFSO	-186.7308	-186.7308	6.4260E-08
	改进 SFSO	-186.7309	-186.7309	2.7356E-10
F2	SFSO	0.0015	3.1541E-05	5.4582E-03
	改进 SFSO	5.3002E-10	3.3533E-11	2.7938E-10
F3	SFSO	5.2231E-05	3.5634E-06	1.2863E-05
	改进 SFSO	4.4163E-10	1.5854E-12	2.7816E-10
F4	SFSO	1.2362E-04	4.6150E-05	1.1943E-04
	改进 SFSO	3.3647E-10	6.3194E-13	3.4126E-10
F5	SFSO	2.1645E-04	7.4521E-05	5.1254E-04
	改进 SFSO	9.3422E-10	6.5346E-10	7.4411E-11
F6	SFSO	7.6713E-03	3.1943E-04	6.9241E-03
	改进 SFSO	9.8798E-10	9.5601E-10	1.3298E-11

为了进一步研究改进 SFSO 算法跳出局部最优的能力,选取 30 维 Rastrigin 函数进行测试,通过观察种群中某个行人的受力情况、适应度值以及种群方差变化来说明改进 SFSO 算法跳出局部最优的能力。

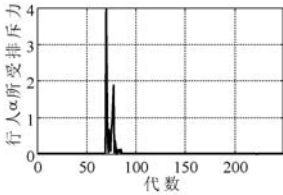


图 1 行人  $\alpha$  所受排斥力

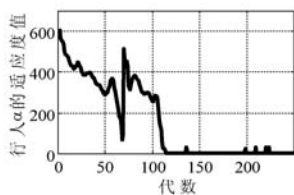


图 2 行人  $\alpha$  的适应度值

图 3 是种群在整个进化过程中的方差图,由图明显看出,在大约 70 代时,当前代种群整体方差水平较低,说明大多数行人均集中在某区域搜索,而从图 2 看出此时适应度值并未达到全局最优值附近,由于行人大量聚集,从图 1 看出同一子群  $G_k$  中其他个体对  $\alpha$  排斥力急剧增大,使子群中的行人重新分布在解空间中搜索,重新分布的行人在改进 SFSO 搜索机制的引导下,使算法逐渐收敛到全局最优,如图 2 所示。综上所述,排斥力可以有效阻止算法陷入局部最优,增强了算法的全局搜索能力,受排斥力作用的个体重新选定目标后,执行改进 SFSO 的搜索机制,再利用行人的单维协作与多维协作相结合,加快了算法的收敛速度并提高了求解精度。

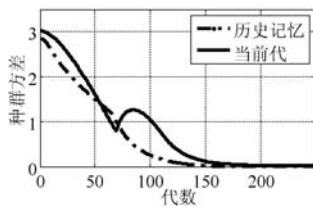


图 3 种群的方差

3.2 SVM 参数优化

3.2.1 参数设置

- GA:  $p_s = 0.2, p_c = 0.8; p_m = 0.1, N = 50, L = 20$ 。
- PSO:  $c_1 = c_2 = 1.49, \omega = 0.9, N = 30$ 。
- 改进 SFSO:  $N = 30, \eta = 10^{-3}, \varepsilon = 5.1, A = 10^{-4}, B = 0.1, R = 1.001, V_{fac} = 5 \times 10^{-6}, \tau = 1, \lambda = 6.25 \times 10^{-3}, \mu = 0.05$ 。

进行归一化预处理。同时,为了说明本文算法的有效性,选择 PSO、GA 这两种经典的群智能优化算法进行对比。

表 3 UCI 数据集说明

数据类别	维数	训练样本	测试样本	类别数
Breast	9	200	77	2
Diabetes	8	468	300	2
Heart	13	170	100	2
Ionosphere	34	200	151	2
Iris	4	90	60	3
Wine	13	89	89	3

### 3.2.3 实验结果与分析

通过对 UCI 标准数据集进行测试,三种方法对不同数据集的分类正确率以及对应的  $C, \sigma$  如表 4 所示。由实验结果可以看出,对于 Breast、Diabetes、Heart、Wine 这 4 个数据集,通过改进 SFSO 算法优化的 SVM 最终得到的分类正确率高于 GA 和 PSO 优化后的结果。由于 GA 与 PSO 会出现不同程度的早熟收敛,且易陷入局部最优,由前面的分析可知,改进 SFSO 借助排斥力的作用,可以有效地避免陷入局部最优,期望力与协作的共同作用,提高了算法的搜索能力以及求解精度,因而得到了更高的分类正确率。虽然对于 Iris,改进 SFSO 算法与 GA 的分类正确率相同,而对于 Ionosphere 数据集,三种方法的分类正确率相同,但是通过每种算法对这两个数据集独立运行 10 次,进一步观察分类准确率的箱线图(如图 4 所示),发现改进 SFSO 算法优化的 SVM 分类稳定性高于其他两种算法。

为了进一步分析三种算法在 SVM 参数组合寻优的收敛性问题,作如图 5 - 图 10 三种算法对 6 个数据集的分类对比图。

由图 5 - 图 10 看出,通过对六个 UCI 数据集进行分类测试并选取最优的 SVM 参数组合的实验表明:基于改进 SFSO 算法的 SVM 参数寻优比基于 GA、PSO 的参数寻优具有更好的分类准确率,同时具有更快的收敛速度。综合以上实验可以看出,基于改进 SFSO 算法的 SVM 参数寻优法具有更好的学习性能。

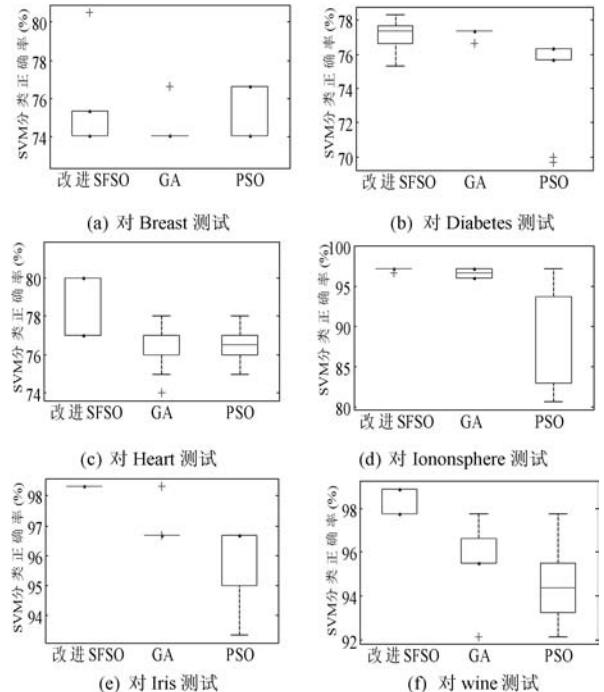


图 4 三种算法对六个数据集分类准确率箱线图

为了验证改进 SFSO 算法优化的 SVM 对不同数据分类的准确性,本文选取 UCI 标准数据集中 6 个代表性的数据进行训练和测试,各类数据的属性如表 3 所示。实验中,为了消除不同量纲之间的数据差异对 SVM 预测精度和推广能力的影响,对数据

表 4 三种方法对六个数据集的最优分类结果对比表

数据类别	算法	C	$\sigma$	分类正确率 (%)
Breast	改进 SFSO	7.1	0.07	80.52
	GA	49.34	0.01	76.62
	PSO	7.43	0.12	76.62
Diabetes	改进 SFSO	16.78	0.23	78.33
	GA	17.53	0.29	77.33
	PSO	13.91	1.13	76.33
Heart	改进 SFSO	0.15	0.11	80
	GA	0.13	0.14	78
	PSO	29.31	0.01	78
Ionosphere	改进 SFSO	1.11	0.66	97.14
	GA	0.74	0.65	97.14
	PSO	12.91	0.08	97.14
Iris	改进 SFSO	1.07	1.91	98.33
	GA	5.24	0.36	98.33
	PSO	5.86	30.24	96.67
Wine	改进 SFSO	39.94	1.34	98.88
	GA	1.03	2.99	97.75
	PSO	5.35	2.83	97.75

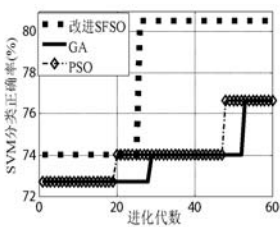


图 5 对 Breast 分类图

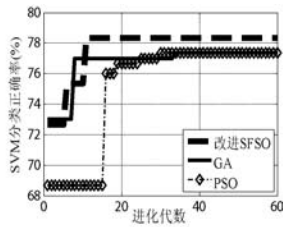


图 6 对 Diabetes 分类图

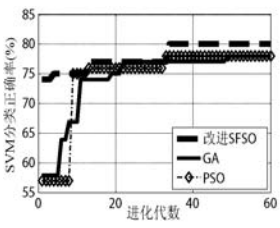


图 7 对 Heart 分类图

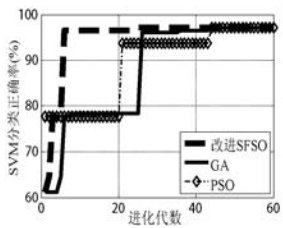


图 8 对 Ionosphere 分类图

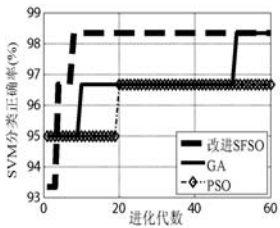


图 9 对 Iris 分类图

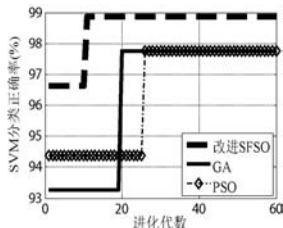


图 10 对 Wine 分类图

## 4 结 语

SVM 的参数选择是一个既有实际价值又有理论意义的研究课题。本文针对 SVM 参数选择的问题提出一种改进基于社会力模型的群智能优化算法的 SVM 参数优化方法。在改进 SFSO 算法特有的搜索机制的引导下,使全局搜索和局部搜索建立了良好的平衡。实验结果表明改进后的 SFSO 算法不仅对于求

解函数优化问题具有较强的鲁棒性和较高的求解精度,而且经改进 SFSO 算法优化后的 SVM 具有更快的收敛速度和更高的分类正确率。

## 参 考 文 献

- [1] Vapnik V. The nature of statistical learning theory[M]. New York: Wiley, 1998.
- [2] 孙继平, 余杰. 基于支持向量机的煤岩图像特征抽取与分类识别[J]. 煤炭学报, 2013, 38(A02): 508-512.
- [3] 徐玉秀, 杨文平, 吕轩, 等. 基于支持向量机的汽车发动机故障诊断研究[J]. 振动与冲击, 2013, 32(8): 143-146.
- [4] 郭烈, 张明恒, 李琳辉, 等. 一种基于支持向量机的行人识别方法研究[J]. 大连理工大学学报, 2011, 51(4): 604-610.
- [5] 徐鹏, 姜凤茹. 基于蚁群优化支持向量机的短时交通流量预测[J]. 计算机应用与软件, 2013, 30(3): 250-254.
- [6] Lhan L, Teze G. A genetic algorithm-support vector machine method with parameter optimization for selecting the tag SNPs[J]. Journal of biomedical informatics, 2013, 46(2): 328-340.
- [7] 于明, 艾月乔. 基于人工蜂群算法的支持向量机参数优化及应用[J]. 光电子·激光, 2012, 23(2): 374-378.
- [8] Chen P W, Wang J Y, Li H M. Model selection of SVMs using GA approach[C]//Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on. IEEE, 2004, 3: 2035-2040.
- [9] Zheng C H, Jiao L C. Automatic parameters selection for SVM based on GA[C]//Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on. IEEE, 2004, 2: 1869-1872.
- [10] 邵信光, 杨慧中, 陈刚. 基于粒子群优化算法的支持向量机参数选择及其应用[J]. 控制理论与应用, 2006, 23(5): 740-743.
- [11] Helbing D, Molnar P. Social force model for pedestrian dynamics[J]. Physical review E, 1995, 51(5): 4282-4286.
- [12] 阎高伟, 李闯勤, 石兵, 等. 基于社会力模型的群体优化算法[J]. 控制工程, 2012, 19(6): 1238-1243.
- [13] 邓乃扬, 田英杰. 数据挖掘中的新方法: 支持向量机[M]. 北京: 科学出版社, 2004.
- [14] Civicioglu P. Artificial cooperative search algorithm for numerical optimization problems[J]. Information Sciences, 2013, 229: 58-76.

(上接第 222 页)

- [9] 李东炜, 刘镇章. 基于 Android 智能手机的实验室预约系统设计[J]. 实验技术与管理, 2012, 29(12): 192-202.
- [10] 黄艺锋, 闫巧. 基于 Android 平台电子词典的设计与实现[J]. 计算机应用, 2011, 31(Z2): 228-232.
- [11] 黄婧, 张林, 季刊, 等. 基于 Android 平台的 GPRS 流量统计系统设计[J]. 计算机系统应用, 2011, 20(12): 141-143.
- [12] 姚昱旻, 刘卫国. Android 与 J2ME 平台间即时通信的研究与实现[J]. 计算机系统应用, 2008, 17(12): 118-120.
- [13] 姚昱旻, 刘卫国. Android 的架构与应用开发研究[J]. 计算机系统应用, 2008, 17(11): 110-112.
- [14] 张涛, 黄强, 毛磊雅, 等. 一个基于 JSON 的对象序列化算法[J]. 计算机工程与应用, 2007, 43(15): 98-100.
- [15] 裴佳迪, 马超, 孙仁贵. Android 应用开发全程实录[M]. 北京: 人民邮电出版社, 2012.
- [16] 郭宏志. Android 应用开发详解[M]. 北京: 电子工业出版社, 2010.