

基于改进 ELM 的撑杆跳起跳过杆轨迹模型研究

汪亚明^a, 李斌权^a, 韩永华^a, 马 可^b

(浙江理工大学, a. 信息学院; b. 体育部, 杭州 310018)

摘 要: 目前对起跳过杆的研究仅仅专注于某一个环节,并未考虑到其他环节间的影响,为此对运动员关键部位进行标记,利用三维红外运动捕捉系统对整个起跳过杆过程进行研究。针对起跳过杆的特点,提出了一种改进的极限学习机算法。首先根据结构风险最小化理论对极限学习机进行正则化;然后利用差分进化算法的全局寻优能力对极限学习机隐层参数的选取进行优化,提高精度;最后建立起跳过杆模型。撑杆跳高的实验结果表明,该算法能有效克服极限学习机易出现过拟合的缺点,具有更好的泛化能力。仿真实验结果表明该方法有效。

关键词: 起跳过杆;极限学习机;差分进化算法;正则化;运动轨迹

中图分类号: TP399

文献标志码: A

文章编号: 1673-3851 (2018) 01-0075-07

0 引 言

在撑杆跳高运动中,起跳、团身、过杆是几个比较重要的基础技术^[1]。要想成功越过横杆,运动员必须通过助跑起跳获得一定的动能,在团身阶段完成空中动作以致过杆。我国撑杆跳高运动成绩目前提高较快,但整体水平一直落后于世界,在亚洲的成绩排名也不理想。所以提升起跳、团身、过杆技术成为了提高撑杆跳高成绩非常重要的方面。

Fukushima 等^[2]利用撑杆跳高机器人,论证了在杆上更积极主动地团身,能改善撑杆跳高成绩。Liu 等^[3]提出了通过动力学优化控制模型,结合测量的相关参数,来对撑杆跳高运动员的表现做分析,发现握杆高度、起跳角度以及撑杆的柔性度与撑杆跳高成绩的关系。Nishikawa 等^[4]利用机器人论证了撑杆跳高过程中恰当的摆体时机能提高撑杆跳高的运动成绩。李北玉^[5]通过分析文献资料,指出决定撑杆跳高成绩的主要因素有运动员握杆的高度、起跳时的速度和腾起的高度。上述对撑杆跳高的研究要么只考虑了运动的某一个环节,忽略了其他环

节间的相互影响和人体在杆上具体的动作形态;要么只是根据数理统计方法得出影响运动成绩的因素,而没有给出帮助运动员提升日常训练的手段。针对上述问题,本文在分析了原始极限学习机(Extreme learning machine, ELM)隐层节点参数选择方法的基础上,通过正则化理论以及差分进化算法(Differential evolution algorithm, DE)对 ELM 进行参数寻优,提出了 DE-RELM 起跳过杆模型。该模型充分考虑了运动员起跳、团身、过杆三个环节,并提取相应特征作为此次起跳过杆轨迹的特征,以此来判定过杆成功与否。同时对过杆成功与失败时提取的特征进行对比分析,以帮助运动员提升日常训练,实现对起跳过杆过程的指导。

1 起跳过杆模型算法原理

1.1 ELM 算法原理

由 Huang 等^[6-7]提出的 ELM 算法,是一种快速的单隐层神经网络训练算法,可用于进行分类和回归预测分析^[8]。它的特点是在网络参数训练过程中,隐层参数随机选取,外权的求解化归成求解一个

收稿日期: 2017-06-20 网络出版日期: 2017-10-10

基金项目: 浙江省自然科学基金重点项目(LZ15F020004);机械工程专业浙江省高校重中之重学科和浙江理工大学重点实验室优秀青年人才培养基金(ZSTUME01B17);浙江理工大学 521 资助项目;浙江省自然科学基金一般项目(LY17F020034)

作者简介: 汪亚明(1972-),男,浙江衢州人,教授,主要从事图像处理模式识别等方面的研究。

通信作者: 韩永华, E-mail: hanyonghuahan@163.com

矩阵的 Moore-Penrose 广义逆问题。

ELM 算法的网络结构如图 1 所示,可以看出,它是一个简单的 SLFN(单隐藏层前馈神经网络)。

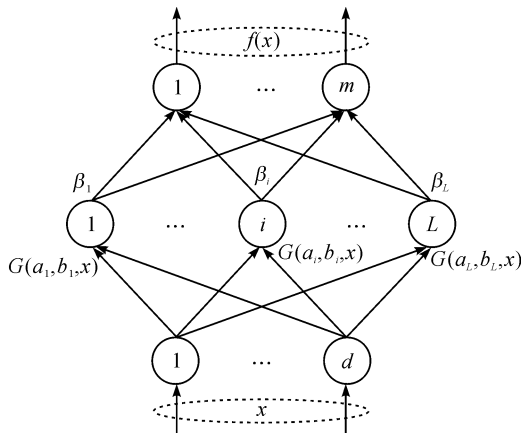


图 1 ELM 网络结构

图 1 中 a_i 和 b_i 为隐层节点参数, β_i 表示第 i 个隐层节点和网络输出之间的连接权值向量, $G(a_i, b_i, \mathbf{x})$ 表示第 i 个隐层节点对应于样本 \mathbf{x} 的隐层节点输出, 隐层节点的激活函数可以是 RBF、Sigmoid 或 Sine 函数等。对于具有 L 个隐层节点的单隐层神经网络的输出表达式如式(1)所示:

$$f_L(\mathbf{x}_j) = \sum_{i=1}^L \beta_i G(a_i, b_i, \mathbf{x}_j) = t_j, \quad j = 1, 2, \dots, N \quad (1)$$

式(1)可以简记为式(2):

$$\mathbf{H}\boldsymbol{\beta} = \mathbf{T} \quad (2)$$

其中 \mathbf{H} 如式(3)所示:

$$\mathbf{H}(a_1, \dots, a_L, b_1, \dots, b_L, \mathbf{x}_1, \dots, \mathbf{x}_N) = \begin{bmatrix} G(a_1, b_1, \mathbf{x}_1) \cdots G(a_L, b_L, \mathbf{x}_1) \\ \vdots \\ G(a_1, b_1, \mathbf{x}_N) \cdots G(a_L, b_L, \mathbf{x}_N) \end{bmatrix}_{N \times L} \quad (3)$$

$\boldsymbol{\beta}$ 和 \mathbf{T} 分别如式(4)和式(5)所示:

$$\boldsymbol{\beta} = [\beta_1^T, \dots, \beta_L^T]_{L \times M}^T \quad (4)$$

$$\mathbf{T} = [t_1^T, \dots, t_N^T]_{N \times M}^T \quad (5)$$

设代价函数 $E(\mathbf{W})$ 表示网络输出和样本的实际值间的误差平方和, 所以问题求解就是寻找最优全值 $\mathbf{W} = (\mathbf{a}, \mathbf{b}, \boldsymbol{\beta})$ 使 $E(\mathbf{W})$ 最小, 其数学模型如式(6)所示:

$$\begin{aligned} \arg \min_{\mathbf{W}=(\mathbf{a}, \mathbf{b}, \boldsymbol{\beta})} E(\mathbf{W}) &= \arg \min_{\mathbf{W}=(\mathbf{a}, \mathbf{b}, \boldsymbol{\beta})} \|\mathbf{e}\|^2 \\ \text{s. t. } \sum_{i=1}^L \beta_i G(a_i, b_i, \mathbf{x}_j) - t_j &= e_j, \quad j = 1, 2, \dots, N \end{aligned} \quad (6)$$

其中 e_j 是第 j 个样本的实际值和网络输出间的误差。

1.2 正则化的 ELM

由统计学理论可知, 实际风险应当由经验风险和置信范围两部分组成^[9]。虽然 ELM 泛化能力强, 但其只考虑了经验风险, 没有考虑置信范围, 可能会导致过拟合问题, 所以本文把结构风险最小化理论引入到 ELM 中, 提出正则化的 ELM (Regularized ELM, RELM)。通过参数 λ 来调节两种风险的比例, 并取得最佳的折中。RELM 数学模型如式(7)所示:

$$\begin{aligned} \arg \min_{\boldsymbol{\beta}} E(\mathbf{W}) &= \arg \min_{\boldsymbol{\beta}} \left(\frac{1}{2} \|\boldsymbol{\beta}\|^2 + \frac{1}{2} \lambda \|\mathbf{e}\|^2 \right) \\ \text{s. t. } \sum_{i=1}^L \beta_i G(a_i, b_i, \mathbf{x}_j) - t_j &= e_j, \quad j = 1, 2, \dots, N \end{aligned} \quad (7)$$

其中: $\boldsymbol{\beta}$ 如式(4)所示; $\|\boldsymbol{\beta}\|^2$ 为约束项, 表示经验风险; λ 表示两种结构的比例参数, 其最佳值可通过交叉验证的方式来确定。上式条件极值问题可利用拉格朗日方程转换为无条件极值问题进行求解, 如式(8)所示:

$$\begin{aligned} \ell(\boldsymbol{\alpha}, \boldsymbol{\beta}, \mathbf{e}) &= \frac{\lambda}{2} \|\mathbf{e}\|^2 + \frac{1}{2} \|\boldsymbol{\beta}\|^2 - \\ &\sum_{j=1}^N \alpha_j (\beta_i G(a_i, b_i, \mathbf{x}_j) - t_j - e_j) = \\ &\frac{\lambda}{2} \|\mathbf{e}\|^2 + \frac{1}{2} \|\boldsymbol{\beta}\|^2 - \boldsymbol{\alpha}(\mathbf{H}\boldsymbol{\beta} - \mathbf{T} - \mathbf{e}) \end{aligned} \quad (8)$$

其中 $\boldsymbol{\alpha} = [\alpha_1, \dots, \alpha_N]$, 表示拉格朗日乘子。

1.3 差分进化算法原理

差分进化算法 (Differential evolution algorithm, DE)^[10] 是一种高效的基于种群迭代的群智能优化算法。该算法作为一种新的进化计算技术, 具有结构简单、待定参数少、不易陷入局部最优、收敛速度快等优点, 在智能计算领域成为了一个研究热点。DE 算法的具体操作如下:

a) 初始化。NP 个个体组成的初始种群 $\mathbf{x}_i(0)$, 维数为 D , $\mathbf{x}_i(0)$ 可随机生成, 如式(9)所示:

$$\{\mathbf{x}_i(0) | x_{j,i}^L \leq x_{j,i}(0) \leq x_{j,i}^U, \quad i = 1, 2, \dots, NP; \\ j = 1, 2, \dots, D\} \quad (9)$$

其中 $x_{j,i}(0)$ 如式(10)所示:

$$x_{j,i}(0) = x_{j,i}^L + \text{rand}(0, 1)(x_{j,i}^U - x_{j,i}^L) \quad (10)$$

其中 $\mathbf{x}_i(0)$ 表示种群中第 0 代第 i 条染色体, $x_{j,i}(0)$ 表示种群中第 0 代第 i 条染色体的第 j 条基因, $x_{j,i}^L$ 和 $x_{j,i}^U$ 分别表示第 i 条染色体的第 j 个基因的下界和上界。rand(0, 1) 表示在 (0, 1) 区间均匀分布的随机数。

b) 变异。变异操作在差异演化中十分重要, 是

从种群中随机选择 3 个个体: \mathbf{x}_{k_1} 、 \mathbf{x}_{k_2} 和 \mathbf{x}_{k_3} , 且 $k_1 \neq k_2 \neq k_3 \neq i$, 则变异操作如式(11)所示:

$$\mathbf{v}_i(g+1) = \mathbf{x}_{k_1}(g) + F \cdot (\mathbf{x}_{k_2}(g) - \mathbf{x}_{k_3}(g)) \quad (11)$$

其中: F 为缩放因子, 决定着偏差向量的方法比例; $\mathbf{x}_i(g)$ 表示种群中第 g 代第 i 条染色体。在进化过程中, 通过判断染色体中的基因是否满足边界条件来保证变异的有效性, 即 $x_{j,i}^L \leq x_{j,i}(g) \leq x_{j,i}^U$, $j = 1, 2, \dots, D$ 。如果不满足, 则用随机方法重新生成基因。

c) 交叉。为了添加种群的多样性, 进行交叉操作, 如式(12)所示:

$$u_{j,i}(g+1) = \begin{cases} v_{j,i}(g+1), & \text{如果 } \text{rand}(0,1) \leq CR \text{ 或者 } j = j_{\text{rand}} \\ x_{j,i}(g), & \text{其它} \end{cases} \quad (12)$$

其中 $CR \in [0,1]$ 为交叉概率, j_{rand} 为 $[1, 2, \dots, D]$ 上的随机数。

d) 选择。为选择进入下一代种群的个体, DE 算法用适应度函数 f 对向量 $\mathbf{u}_i(g+1)$ 和 $\mathbf{x}_i(g)$ 进行比较来生成 $g+1$ 代种群, 如式(13)所示:

$$\mathbf{x}_i(g+1) = \begin{cases} \mathbf{u}_i(g+1), & \text{如果 } f(\mathbf{u}_i(g+1)) \leq f(\mathbf{x}_i(g)) \\ \mathbf{x}_i(g), & \text{其它} \end{cases} \quad (13)$$

e) $g = g+1$ 。反复执行步骤 b) 到步骤 e), 直到满足所要求的收敛精度或达到最大的进化代数。

1.4 改进的 RELM

虽然 RELM 考虑了结构风险, 且在网络参数的确定过程中, 隐节点参数是随机选取的, 可以提高算法的学习速度, 但相对于输入数据而言这些参数并不是最优的选择, 精度也有待提高。在实际应用中, 为使得 ELM 有较好的性能, 往往会增加隐层节点的个数, 这样做会带来过拟合的风险, 同时也会增加算法的复杂性。为提高算法的精度, 减少隐层的节点数, 本文利用差分进化算法的全局寻优能力, 为正则化的极限学习机寻找合适的网络参数, 并将其作为训练 DE 算法的适应度指标, 以获得最佳网络。最后将优化的 ELM 应用到数据的分类处理当中^[11-12]。

利用差分进化优化的正则化极限学习机 (DE-RELM) 流程如图 2 所示, 基本步骤如下:

给定一个训练样本集 $S = \{(\mathbf{x}_i, \mathbf{t}_i) \mid \mathbf{x}_i \in \mathbf{R}^n, \mathbf{t}_i \in \mathbf{R}^m, i = 1, 2, \dots, N\}$, 激励函数 $g(x)$, 最大迭代次数 G_{max} 以及隐层节点的个数 k 。其中, 每个样本表示一次起跳过杆的特征向量和是否成功过杆。

a) 采用经验值估计法初始化参数 NP 、 F 、 CR ,

并随机生成一组初始种群 $\mathbf{Q}_{NP \times D}$, 每种种群 $\mathbf{Q}(t) = \{a_{11}, a_{12}, \dots, a_{1l}, a_{21}, a_{22}, \dots, a_{2l}, \dots, a_{n1}, a_{n2}, \dots, a_{nl}, b_1, b_2, \dots, b_l\}$ 。

b) 对于每个种群, 计算隐层输出矩阵得到输出权值, 并求出其相应的均方根误差作为适应度指标, 具体如式(14)所示:

$$J = \sqrt{\frac{\sum_{j=1}^N \sum_{i=1}^L \|\beta_i G(a_i, b_i, \mathbf{x}_j) - \mathbf{t}_j\|^2}{N \times L}} \quad (14)$$

c) 种群中的个体根据 DE 算法, 先进行变异得到个体 $\mathbf{V}_i(g+1)$, 再进行交叉获得新个体 $\mathbf{U}_i(g+1)$, 然后选取有竞争力的个体进入下一代种群的个体, 选取过程如式(15)所示:

$$\mathbf{Q}_i(g+1) = \begin{cases} \mathbf{U}_i(g+1), & J(\mathbf{U}_i(g+1)) \leq J(\mathbf{Q}_i(g)) \\ \mathbf{Q}_i(g), & \text{其它} \end{cases} \quad (15)$$

其中, $J(\cdot)$ 代表其适应度指标。

d) 根据迭代次数判断程序是否继续进行。若在迭代次数范围内则重复步骤 b) 和步骤 c), 直至目标完成, 否则直接进入步骤 e)。

e) 输出权值 β 。

f) 程序结束。

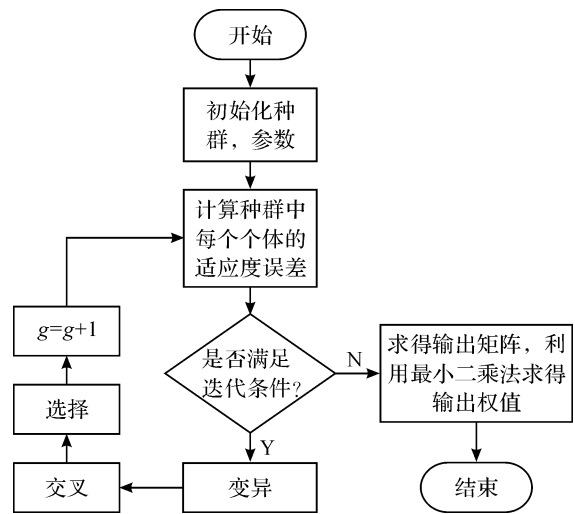


图 2 DE-RELM 算法流程

2 仿真实验和分析

2.1 三维红外运动捕捉系统

OptiTrack 三维红外运动捕捉系统通过在人体或其他物体上布置标记点, 由高速相机接受标记点反射的红外光, 然后识别标记点来认知人体, 从而实时捕获人体的运动姿态和在空间坐标系中位置信息。图 3(a)为附着标记点的人体。



(a)附着标记点的人体 (b)在关键部位贴上标记点的运动员

图3 带有标记点的人体

本文使用的高速相机为12台Prime13(简称p13),其最高帧速为240 fps,分辨率达到 1280×1024 ,视场角为 56° ,处理延迟为4.2 ms,最远工作距离为14 m。为了避免标记点越出高速相机的捕捉范围,导致检测数据可能出现的空白,应尽可能扩大捕捉范围,防止出现错误。

2.2 实验数据采集

本实验的研究对象是撑杆跳过程中的起跳过杆,即起跳、团身、过杆,这是撑杆跳运动中三个重要的阶段。实验要求运动员在保证最大过杆率的情况下,尽可能越过高的横杆,取得好的成绩。实验开始前,告知运动员实验目的和注意事项,运动员有5 min用于热身。整个实验过程中运动的质量和结果不反馈给运动员。为了更好地研究运动员撑杆跳的完成情况,在运动员的两髌、左脚踝、左膝盖附着标记点,如图3(b)所示。左脚踝和左膝盖处的标记点的位置信息用来描述运动员在团身即后仰举腿阶段,双腿的弯曲程度以及与身体的距离。由于在运动过程中可能会发生身体重心标记点被遮挡的现象,无法对重心直接进行跟踪,所以本文选择标记两髌,用其连线的中点近似人体重心。采集样本的过程中,如果没有完成过杆动作或是横杆被触碰落地,认为是过杆失败;否则记为成功过杆。每次的撑杆跳结果按序记录于文件中,且结果、质量不向运动员进行反馈。实验共采集200组数据样本,其中成功过杆153次,过杆失败47次。随机选取115组成功过杆的样本和35组过杆失败的样本,共150组构成一个训练集用于训练,剩余50组用于测试,并反复进行10次,共构成10组实验数据集。图4所示为一组实验数据集的数据构成。实验全程使用同一个撑杆、横杆。

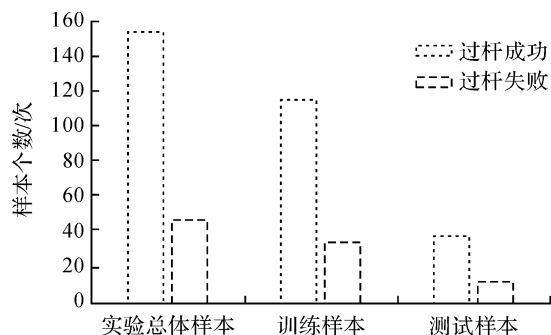


图4 样本数据统计

2.3 ELM网络参数设置

2.3.1 输入向量和输出向量

根据摄像机记录的两髌的运动轨迹来求得两髌中心点的运动轨迹,以此来近似拟合人体重心的运动轨迹。“自由起跳”的理论基础是:为获得良好的摆体效果,运动员起跳时的水平速度应尽量保持,并将这一速度转化到后续的空中动作中^[13]。而腾起角可以间接反映“自由起跳”技术掌握的情况。图5是运动员助跑最后阶段到摆体前脚踝的Y轴运动轨迹,通过分析可以确定起跳前的最后一个极小值点(T_1)就是起跳时刻,设此时脚踝坐标为 $(x_{\min}, y_{\min}, z_{\min})$,并记录起跳后y轨迹达到第一个极大值点的坐标 $(x_{\max}, y_{\max}, z_{\max})$;再利用运动员重心运动轨迹,如图6所示,设起跳时身体重心坐标为 (x_1, y_1, z_1) ,计算紧随起跳点后的两帧重心位置在世界坐标系X、Y、Z方向的位移和两帧的时间间隔(两帧的时间间隔为帧速的倒数)的比值,作为起跳点处运动员在Y轴方向的速度 v_1^y ,X轴方向的速度 v_1^x 和Z轴上的速度 v_1^z (其他阶段的速度计算同理);按式(16)计算起跳点对应的腾起角 θ_1 :

$$\theta_1 = \arccos \sqrt{\frac{(x_{\max} - x_{\min})^2 + (z_{\max} - z_{\min})^2}{(x_{\max} - x_{\min})^2 + (y_{\max} - y_{\min})^2 + (z_{\max} - z_{\min})^2}} \quad (16)$$

并将 v_1^x, v_1^y, v_1^z 和 θ_1 作为起跳时的特征。团身阶段,要求运动员双腿弯曲并向胸部靠拢,背部几乎与地面平行。本文通过计算左膝盖处的标记点与人体重心的距离,在运动员腾空阶段,该距离达到极小值时,表明运动员正处于团身状态,利用左髌和左膝的连线以及左脚踝和左膝的连线,根据余弦定理来计算此时人体的左膝角 θ_2 ,并选取 θ_2 和重心的速度 $v_2 = \{v_2^x, v_2^y, v_2^z\}$ 以及重心位置信息 $p_2 = (x_2, y_2, z_2)$ 作为团身阶段的特征,其中 x_2, y_2, z_2 分别表示人体在X轴、Y轴和Z轴的位置,速度和位置信息可以体现出起跳的早晚^[14]。过杆阶段从身体推离过杆开始。

好的过杆技术应是身体重心达到腾跃最高点时,重心正好处于横杆的正上方。由图 6 身体重心运动轨迹可以得出,撑杆跳高过程中最后一个极大值点(T_2)就是运动员身体重心达到腾跃最高点的时刻,记录此时身体重心坐标为 $p_{3m} = (x_{3m}, y_{3m}, z_{3m})$,并将该时刻身体重心的速度 $v_3 = \{v_3^x, v_3^y, v_3^z\}$ 、身体重心与横杆的水平距离 l_x 和 x_{3m} 比值 l_{x0} 、垂直距离 l_y 和 y_{3m} 比值 l_{y0} 作为过杆阶段的特征。为了方便处理,对 p_2 进行以下归一化处理的得到 p_{20} ,如式(17)所示:

$$p_{20} = (x_{20}, y_{20}, z_{20}) = \left(\frac{x_2}{x_{3m}}, \frac{y_2}{y_{3m}}, \frac{z_2}{z_{3m}} \right) \quad (17)$$

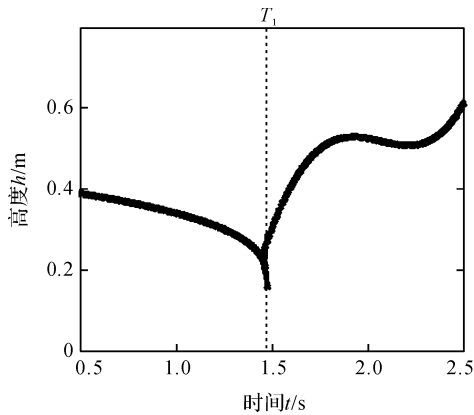


图 5 助跑最后阶段到摆体前脚踝 Y 轴轨迹

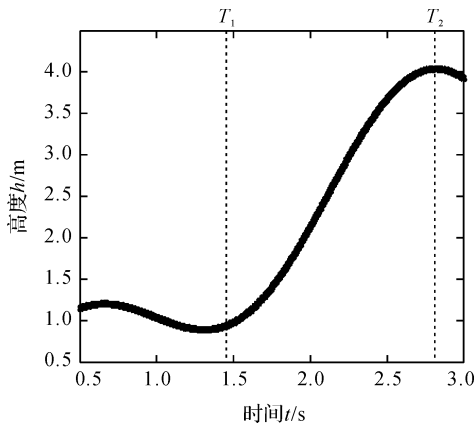


图 6 身体重心 Y 轴轨迹

根据上述记录的信息,最终构成特征向量 $X = \{v_1, \theta_1, v_2, \theta_2, p_{20}, v_3, l_{y0}, l_{x0}\}$,所以模型的输入向量是一个 16 维的向量。对实验样本进行特征提取形成样本数据 $T = \{(X_i, Y_i)\}$, $i = 1, 2, \dots, 200$,其中输入向量 $X_i \in \mathbf{R}^{16}$ 为第 i 组实验样本的提取的特征,输出向量为 $Y_i \in \{-1, 1\}$,当 X_i 是成功过杆提取的特征时, Y_i 取 1,否则取 -1。

2.4 实验结果与分析

本文在 Intel Xeon(R) E5-2620 v3 CPU 2.40 GHz, 32.0 GB 内存的机器上采用 Matlab R2012a,实现了 DE-RELM 起跳过杆模型。本实验是在 10 组实验数

据集下完成的以提高算法说服力。为方便讨论,本实验将 DE 算法中 CR 、 F 、 NP 以及迭代次数固定。经过大量实验最终本文选取 $CR=0.6$, $F=0.8$ 。虽然搜索到全局最优解的概率会随着种群规模的增加而变大,但所需的计算量和计算时间也会随之增加,一般取 100~300,本文取 $NP=200$ 。在不影响算法性能的前提下,算法的最大迭代次数设为 500。

隐层节点数对算法的影响见表 1。从表 2 中可以看出,当迭代次数一定时,随着隐层节点数的增加,算法误差逐渐减低。当隐层节点数为 10 时,DE-RELM 的误差率远小于 ELM,当隐层节点数为 15 时,算法误差率虽有减小但变化不大,为减少算法计算量,选择隐层节点数为 10 的 DE-RELM 分类器与传统 ELM 和 SVM 支持向量机进行对比。为了提高算法的说服力,每个算法都执行 10 次,并记录平均值。

表 1 隐层节点数对算法的影响

隐层 节点数	训练时间/s		训练误差		测试误差	
	ELM	DE-RELM	ELM	DE-RELM	ELM	DE-RELM
1	0	5.46	0.510	0.5100	0.480	0.490
5	0.19	10.51	0.270	0.2400	0.210	0.180
10	0.29	17.37	0.014	0.0070	0.063	0.004
15	0.47	27.49	0.011	0.0053	0.052	0.003

注:迭代次数为 500 次。

对 10 组实验数据集进行实验的结果见表 2。可以看出,在迭代次数和隐层节点数一定时,DE-RELM 算法的误差率更低,实验结果理想。

表 2 三种算法性能对比

算法	训练时间/s	训练误差	测试误差
SVM	2.31	0.008	0.058
ELM	0.29	0.014	0.063
DE-RELM	17.37	0.007	0.004

对 50 组测试样本分别使用 ELM、SVM、DE-RELM 三种算法进行分类,分类结果对比如图 7 所示。从各算法分类正确率可以看出每个分类器的分类效果。

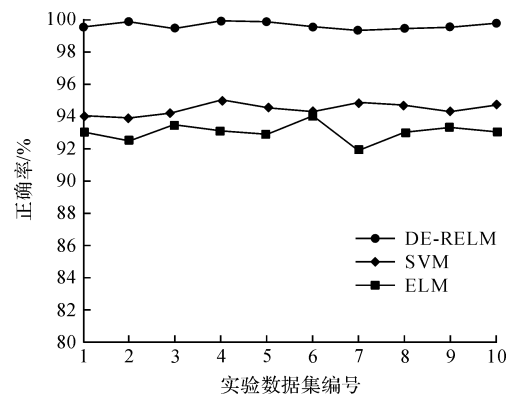


图 7 各分类算法分类结果对比

10组实验数据集下不同测试样本数的平均分类准确率如图8所示。其中,横坐标为测试样本数,纵坐标为分类算法的正确率。

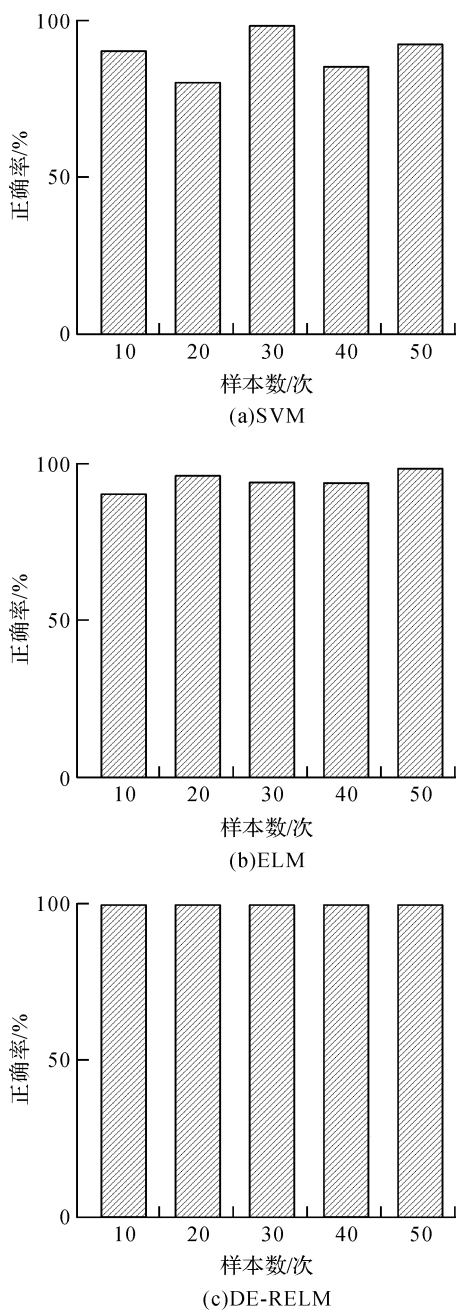


图8 各分类算法正确率分布

从图8可以看出三种分类算法对测试样本的分类情况。随着测试样本数量的增加,SVM和ELM的分类准确率都较高,泛化能力也都还好。但比较而言,DE-REL的性能更佳,泛化能力更强,其分类准确率达到99.61%,达到了实验预期的结果。

上述结果表明,本文提出的DE-REL算法能够对撑杆跳高运动员是否成功过杆进行准确的分类、预测,且效果明显优于ELM、SVM算法。达到

相同或更高的精度时,DE-REL网络结构更优,泛化能力更好。

3 结 语

本文提出的DE-REL模型能帮助提高撑杆跳高运动员起跳过杆技术。通过分析起跳过杆时提取的特征向量,并以此为参考依据进行训练。为提高模型的泛化能力,提升算法的性能,本文将差分进化算法和正则化的极限学习机进行了有机结合。实验表明,基于DE-REL的起跳过杆模型性能远远优于ELM,具有很好的预测的能力,达到了预期目标,可以对运动员进行有效训练的指导。由仿真实验结果可知,本文提出的基于DE-REL的撑杆跳起跳过杆模型有效。

参考文献:

- [1] 陈庆杰,单吉谭. 中国优秀女子撑竿跳高运动员技术研究现状[J]. 辽宁体育科技,2005,27(3):70-71.
- [2] Fukushima T, Nishikawa S, Kuniyoshi Y. Active bending motion of pole vault robot to improve reachable height [C]//2014 IEEE International Conference on Robotics and Automation(ICRA). IEEE,2014:4208-4214.
- [3] Liu G, Nguang S K, Zhang Y. Pole vault performance for anthropometric variability via a dynamical optimal control model[J]. Journal of Biomechanics,2011,44(3):436-441.
- [4] Nishikawa S, Fukushima T, Kuniyoshi Y. Effective timing of swing-up motion by a pole-vaulting robot [C]// International Conference on Advanced Robotics. IEEE, 2013:1-6.
- [5] 李北玉. 对影响撑竿跳高运动成绩的主要因素探讨[J]. 首都体育学院学报,2001,13(1):61-63.
- [6] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: a new learning scheme of feedforward neural networks [C]//IEEE International Joint Conference on Neural Networks,2004. Proceedings. IEEE,2004,2:985-990.
- [7] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: Theory and applications[J]. Neurocomputing, 2006,70(1-3):489-501.
- [8] 邓万字,郑庆华,陈琳,等. 神经网络极速学习方法研究 [J]. 计算机学报,2010,33(2):279-287.
- [9] 吴登国,李晓明. 基于极限学习机的配电网重构[J]. 电力自动化设备,2013,33(2):47-51.
- [10] Storn R, Price K. Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces[J]. Journal of Global Optimization, 1997, 11 (4):341-359.
- [11] 徐涛,郭威,吕宗磊,等. 基于快速极限学习机和差分进

- 化的机场噪声预测模型[J]. 电子与信息学报, 2016, 38(6):1512-1518.
- [12] Chen S, Wu M, Zhao S. Analog circuit fault diagnosis based on DE OS-ELM [C]//Seventh International Symposium on Computational Intelligence and Design. IEEE, 2014:509-513.
- [13] 张智敏, 孙南, 张曼, 等. 撑杆跳高“自由起跳”技术评价的运动学参数分析[J]. 体育科技文献通报, 2012, 20(12): 25-28.
- [14] 陈庆杰, 吕季东. 我国优秀女子撑竿跳高运动员悬垂摆体技术的运动学特征[J]. 成都体育学院学报, 2007, 33(1): 66-69.

Research on trajectory model of pole vault based on improved extreme learning machine

WANG Yaming^a, LI Binqun^a, HAN Yonghua^a, MA Ke^b

(a. School of Information Science and Technology; b. Department of Physical Education, Zhejiang Sci-Tech University, Hangzhou 310018, China)

Abstract: Recently, most of the researches on pole vault only focus on one part, but fail to take into account of the influence of other parts. So, we marked the vault with marker point at the key parts and used the three-dimensional infrared motion capture system to research the progress of pole vault. According to the characteristics of pole vault, the paper presents an improved extreme learning machine algorithm. Firstly, the theory of structural risk minimization was used to regulate extreme learning machine. Then, global optimization ability of differential evolution algorithm was adopted to optimize the selection of hidden layer parameters of extreme learning machine, which aims to improve the accuracy. After that, the model of pole vault was established. Experimental results of pole vault showed that the algorithm can efficiently overcome the overfitting shortcoming of extreme learning machine and has better generalization performance, which proves the effectiveness of the algorithm.

Key words: pole vault; extreme learning machine; differential evolution algorithm; regularization; trajectory

(责任编辑: 康 锋)