

比例延迟微分方程的极限学习机算法

李佳颖, 陈浩

重庆师范大学数学科学学院, 重庆 401331

摘要:目的 针对比例延迟微分方程,提出一种基于极限学习机(ELM)算法的单隐藏层前馈神经网络训练方法,并将该方法推广到求解双比例延迟微分系统。方法 首先,构建一个单隐藏层前馈神经网络并随机生成输入权值和隐藏层偏置;然后,通过计算系数矩阵使其满足比例延迟微分方程及其初值条件,将其转化为最小二乘问题,利用摩尔-彭罗斯广义逆解出输出权值;最后,将输出权值代入构建的神经网络便可获得具有较高精度的比例延迟微分方程数值解。结果 通过数值实验与已有方法的结果进行比较,验证了该方法对处理比例延迟微分方程与双比例延迟微分系统的有效性,且随着选取的训练点和隐藏层节点数量增多,所得到的数值解精度和收敛速度也随之增加。结论 ELM算法对处理比例延迟微分方程以及双比例延迟微分系统具有较好的效果。

关键词:前馈神经网络;比例延迟微分方程;极限学习机;双比例延迟微分系统

中图分类号:O241.8 **文献标识码:**A **doi:**10.16055/j.issn.1672-058X.2024.0001.014

Extreme Learning Machine Algorithm for Pantograph Delay Differential Equations

LI Jiaying, CHEN Hao

School of Mathematical Sciences, Chongqing Normal University, Chongqing 401331, China

Abstract: Objective A single hidden layer feed-forward neural network training method based on extreme learning machine (ELM) was proposed for pantograph delay differential equations, and the method was extended to deal with the system of pantograph equations with two delays. **Methods** Firstly, a feed-forward neural network with a single hidden layer was constructed and the input weights and hidden layer bias were randomly generated. Then, by calculating the coefficient matrix to satisfy the pantograph delay differential equation and its initial value conditions, the equation was transformed into a least squares problem, and the output weight was obtained by using the Moor-Penrose generalized inverse solution. Finally, the numerical solution of the pantograph delay differential equation with high precision could be obtained by inputting the output weights into the constructed neural network. **Results** By comparing the results of numerical experiments with those of existing methods, the effectiveness of the proposed method in dealing with pantograph delay differential equations and the system of pantograph equations with two delays was verified. With the increase in the number of selected training points and hidden layer nodes, the accuracy and convergence rate of the numerical solutions were also increased. **Conclusion** The ELM algorithm is effective in dealing with pantograph delay differential equations and the system of pantograph equations with two delays.

Keywords: feed-forward neural networks; pantograph delay differential equation; extreme learning machine; pantograph equation with two delays

1 引言

延迟微分方程是一类广受关注的微分方程。由于

事物运动规律的影响因素不仅包含当前时刻的状态,而且会受到过去某些时刻或时间段的状态影响,因此

收稿日期:2022-12-13 修回日期:2023-03-02 文章编号:1672-058X(2024)01-0106-07

作者简介:李佳颖(1995—),女,重庆万州人,硕士研究生,从事微分方程数值解法研究。

通讯作者:陈浩(1986—),男,教授,博士,从事微分方程数值解法研究。Email:hch@cqnu.edu.cn。

引用格式:李佳颖,陈浩.比例延迟微分方程的极限学习机算法[J].重庆工商大学学报(自然科学版),2024,41(1):106—112.

LI Jiaying, CHEN Hao. Extreme learning machine algorithm for pantograph delay differential equations[J]. Journal of Chongqing Technology and Business University (Natural Science Edition), 2024, 41(1): 106—112.

大部分事物运动规律都存在延迟现象。延迟微分方程中的延迟项,不仅能够展现当前时刻的状态,而且也能展现出过去某些时刻的状态,因此这种特殊形式的微分方程被证明能够更精确地模拟各种客观事物的变化规律。延迟微分方程被广泛应用于生命科学和工程问题的各个领域,如种群动力学、免疫学、流行病模型、神经网络和控制系统^[1-5]。

本文将考虑一类特殊形式的延迟微分方程—比例延迟微分方程。其中“比例项”一词由奥肯登和泰勒^[6]在 1971 年首次引入,用于研究电力机车的受电弓如何收集电流,此后比例延迟问题得到了众多学者的关注。比例延迟微分方程被广泛应用于经济、控制、数论、电动力学、非线性动力学系统等不同学科领域。

广义比例延迟微分方程初值问题的一般形式为

$$y^{(n)}(t) + a(t)y^{(n)}(p_n t) + \sum_{k=0}^{n-1} b_k(t)y^{(k)}(p_k t) + c(t)y(t) = f(t), t_0 < t \leq T$$

$$y^{(k)}(t_0) = \lambda_k, \lambda_k \in \mathbf{R} (k = 0, 1, \dots, n-1) \quad (1)$$

其中, $a(t)$ 、 $b_k(t)$ 和 $f(t)$ 是解析函数, $p_k \in (0, 1)$ ($k = 0, 1, \dots, n$)。

一直以来,许多学者对比例延迟微分方程的数值解法进行分析研究,其中一些方法显示出很大的潜力。例如, Wang 等^[7]将龙格-库塔方法应用于一类非线性中立型比例延迟微分方程; Yüzbaşı 等^[8]提出一种基于残差校正技术的勒让德配置方法,利用算子方程的残差函数,构造了一个误差微分方程,并对使用勒让德配置法得到的多比例延迟微分方程数值解进行了修正; Doha 等^[9]提出一种基于雅可比有理函数和高斯积分公式相结合的谱雅可比有理-高斯配点法,得到了多比例延迟微分方程的半解析解; Cakmak 等^[10]使用配点法和斐波那契多项式矩阵将比例延迟微分方程问题简化为一个非线性代数系统,然后计算近似解函数的未知系数,进而得到非线性比例延迟微分方程的数值解; Bahgat^[11]应用拉普拉斯变换与变分迭代法求解中立型比例延迟微分方程,该算法只需经过几次迭代即可得到高精度的数值解。上述方法都能获得较高精度的数值解,然而这些方法通常需要离散方程,且都包含大量的计算,存在一定缺陷。

近年来,随着人工智能的大力发展,神经网络算法因其可以避免传统数值方法中的一些缺点,逐渐成为求解微分方程数值解的一种替代方法。根据通用近似定理可知,神经网络是一种通用近似器,它们可以被用于近似任何连续微分方程的解。不仅如此,使用神经网络近似微分方程的解不需要对方程进行离散化处理,由此可以避免离散过程中产生的误差。神经网络方法在求解微分方程方面具有以下几个优点:不需要通过离散微分方程来求进行解;所得到的解是连续的,

而不是在离散点上的数值解;类似的方法可用于不同类型微分方程的初边值问题,且往往具有较高的精度。

神经网络方法也吸引了许多学者对微分方程数值解的研究。Mall 等^[12]利用切比雪夫正交多项式构建了切比雪夫神经网络,将切比雪夫多项式作为输入权值,不仅大幅降低了神经网络所需的参数,而且能够获得具有较高精度的数值解; Sun 等^[13]提出一种基于伯恩斯坦多项式的神经网络模型用于求解微分方程,该方法将伯恩斯坦多项式扩展输入来消除神经网络隐藏层。根据最近的文献,一些作者已经开始使用神经网络方法来研究延迟微分方程。Raja^[14]利用人工神经网络、模拟退火、模式搜索、遗传算法、有源集算法的混合技术求解二阶比例延迟微分方程边值问题; Khan 等^[15]利用前馈神经网络结合列文伯格-马夸尔特算法和贝叶斯正则化通过反向传播算法求解比延迟微分方程; Sabir 等^[16]将分数阶 Meyer 小波神经网络与遗传算法全局搜索优化和序列二次规划局部搜索相结合,给出了分数阶比例莱恩-埃姆登模型的数值解。然而以上神经网络算法都有一个常见问题,即都是利用计算成本较高的数值优化算法来逼近解。这些算法大多需要计算梯度,有时还需要计算误差函数的黑森矩阵,不仅需要进行多次迭代来优化调节参数,而且还会导致训练过程中出现大量的运算,非常耗时。这些研究工作的另一个问题是,它们很少或完全没有强调讨论算法的收敛性和收敛速度。Panghal 等^[17]通过单隐藏层的前馈神经网络和极限学习机算法得到了具有较高精度的延迟微分系统数值解,同时分别利用浅层和多层极限学习机算法验证了使用极限学习机方法训练神经网络时,单隐藏层神经网络的函数逼近效果最好,但未涉及高阶延迟微分方程的求解。

本文着重讨论了上述两个方面,针对广义比例延迟微分方程构建单隐藏层前馈神经网络并使用极限学习机算法进行训练。该算法将比例延迟微分方程转化为最小二乘问题,通过摩尔-彭罗斯广义逆直接计算出最优神经网络输出权值。不同于传统基于梯度的网络训练方法,该算法完全避免了耗时的网络训练参数优化过程,计算量大幅降低,整个网络训练过程的速度要快得多。同时讨论了该算法取不同数量的训练点和隐藏层节点时的精度,并在此基础上,计算了数值收敛速度。通过数值实验与已有方法的结果进行比较,证明该方法对处理比例延迟微分方程具有较好的效果,并将该方法推广到双比例延迟微分系统的数值求解。

2 神经网络模型构建

神经网络是一种通过模拟人脑神经系统,进行分布式并行信息处理的数学模型。神经网络系统是由神经元通过连接权值连接在一起构成的,而不同的连接方式构成了不同种类的网络结构。本文使用的是前馈

型神经网络,该网络主要由输入层、隐藏层和输出层 3 部分构成,每一层由许多相互连接的神经元构成,在神经网络中,神经元也称为节点。连接权值代表各节点之间连接的强弱,这些连接权值通常在神经网络的训练过程中会进行不断地更新调整。当输入信号由输入层到达隐藏层后,经过隐藏层的激活函数激活后再将结果传递到输出层的节点,最后得到输出结果。

在本文的研究中,构建了一个单隐藏层前馈神经网络用于求解比例延迟微分方程,该网络包含一个输入节点和一个输出节点, w_j 表示输入层到隐藏层的权值, v_j 表示隐藏层到输出层的权值, b_j 表示隐藏层节点的偏置。本文将使用相同的约定。该网络的示意图如图 1 所示。

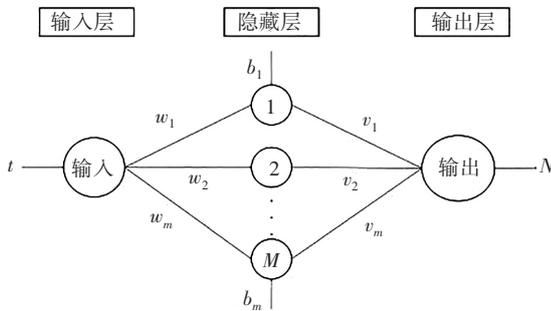


图 1 神经网络图

Fig. 1 Neural network diagram

对于任意的输入 t , 都有输出

$$N = \sum_{j=1}^m v_j \varphi(w_j t + b_j)$$

其中, m 表示隐藏层的节点个数, φ 表示激活函数。本文使用的激活函数为 sigmoid 函数 $s(x) = 1/(1+e^{-x})$, 因为该函数便于求导且自身无穷可微。

定理 1 通用近似定理 (Universal Approximation Theorem): 设 $\varphi(\cdot)$ 是有界、单调递增的连续非常数函数, I_d 为 d 维的单位立方体 $[0, 1]^d$, $C(I_d)$ 是定义在 I_d 上的连续函数集合, 对于任意一个函数 $y \in C(I_d)$ 及 $\varepsilon > 0$, 存在有限和

$$N(t) = \sum_{j=1}^m v_j \varphi(w_j t + b_j)$$

使得

$$|N(t) - y(t)| < \varepsilon, \forall t \in I_d$$

这里 m 为正整数且 $v_j, w_j, b_j \in \mathbf{R}$ 。

根据通用近似定理可知, 只要构造一个单隐藏层前馈神经网络并选取适当的激活函数, 便可以通过简单的神经网络架构去拟合任何连续函数的解。

3 用极限学习机 (ELM) 算法求解比例延迟微分方程

3.1 极限学习机 (ELM) 算法分析

神经网络学习过程有许多不同的训练算法, 最常

见的训练算法是反向传播 (BP) 算法。BP 算法分为正向传播和反向传播两个学习过程。正向传播即对每一个输入信号逐层向前计算最终得到一个输出结果。如果该输出值未达到预想的输出值, 则计算输出值与期望值的误差, 并将该误差从输出层向隐藏层进行反向传播直到输入层。在反向传播的过程中, 会根据误差调整各种参数的值, 不断重复上述过程, 直至误差达到期望值时学习结束。BP 算法的本质是梯度下降法, 这个过程通常需要经过数次迭代, 运算时间较长。

极限学习机 (ELM) 算法^[18] 作为一种新的前馈神经网络训练方法, 已经成为机器学习训练过程中的替代方案。ELM 算法是一种无梯度下降过程训练前馈神经网络的算法。该算法的核心思想是随机生成网络的输入权值和隐藏层偏置, 并且在训练过程中这些参数一直保持不变, 只优化调节训练点和隐藏层节点的个数, 输出权值则通过最小化平方损失函数, 将其转化为最小二乘法问题, 利用 Moore-Penrose 广义逆解出输出权值。其中隐藏层节点数的选择是重点, 如果数量过少, 训练过程中提取到的信息量不足, 网络将无法概括和反映数据的内在规律, 映射能力不强, 出现欠拟合问题; 但数量过多, 又会使网络结构变得复杂, 不仅增加训练时间, 而且容易引起过度拟合, 影响网络的整体性能。与其他传统的基于梯度下降的前馈神经网络算法相比, ELM 算法不仅避免了训练过程中的大量迭代调整, 学习速度极快, 在选择好合适的激活函数和确定隐藏层节点个数后, 算法学习过程中不再需要人为调整参数, 所得解是唯一最优解且收敛快速, 具有良好的泛化性等优势。

3.2 比例延迟微分方程的求解

对于任意一个连续函数 $y(t), t \in [t_0, T]$, 使用含有 m 个隐藏层节点的上述神经网络进行近似, 表达式写为

$$N(t) = \sum_{j=1}^m v_j \varphi(w_j t + b_j) \approx y(t) \tag{2}$$

用 $N(t)$ 替换式 (1) 中的 $y(t)$, 得到:

$$N^{(n)}(t) + a(t_i)N^{(n)}(p_n t) + \sum_{k=0}^{n-1} b_k(t)N^{(k)}(p_k t) + c(t)N(t) = f(t) \tag{3}$$

再将

$$N^{(k)}(t) = w_j^k \varphi^{(k)}(w_j t + b_j), k = 1, 2, \dots, n$$

代入式 (3) 得到:

$$\sum_{j=1}^m v_j \{ w_j^n [\varphi^{(n)}(w_j t + b_j) + a(t_i) \varphi^{(n)}(w_j p_n t + b_j)] + \sum_{k=0}^{n-1} b_k(t) w_j^k \varphi^{(k)}(w_j p_k t + b_j) + c(t_i) \varphi(w_j t + b_j) \} = f(t)$$

最后对 $[t_0, T]$ 上的 s 个点 (t_1, t_2, \dots, t_s) 进行训练, 进一

步得到

$$\sum_{j=1}^m v_j \{ w_j^n [\varphi^{(n)}(w_j t_i + b_j) + a(t_i) \varphi^{(n)}(w_j p_n t_i + b_j)] + \sum_{k=0}^{n-1} b_k(t_i) w_j^k \varphi^{(k)}(w_j p_k t_i + b_j) + c(t_i) \varphi(w_j t_i + b_j) \} = f(t_i) \quad (i = 1, 2, \dots, s) \quad (4)$$

式(4)表示关于 v_j 的线性方程组。

此外,神经网络还需满足初值条件,即

$$\sum_{j=1}^m v_j [w_j^k \varphi^{(k)}(w_j t_0 + b_j)] = \lambda_k \quad (k = 1, 2, \dots, n)$$

则将得到 $s+k$ 个线性方程组,这个线性方程组可以写成矩阵形式

$$Hv = \beta$$

其中, H 是系数矩阵, $\beta = [f(t_1), \dots, f(t_s), \lambda_1, \dots, \lambda_k]^T$ 。

根据 ELM 算法,首先随机生成输入权值 w_j 和偏置 b_j ,计算出矩阵 $H\beta$ 是已知的,然后通过公式 $v = H^{-1}\beta$ 求解输出权值,且 v 是唯一解,最后将得到的 $v = [v_1, v_2, \dots, v_m]^T$ 代入式(2)即可得到比例延迟微分方程的近似解。

大多数实际情况中,矩阵 H 通常既不是方阵又不可逆,求解可能会出现问题,这时可以使用 Moore - Penrose 广义逆来进行求解。

定义 1 若 $n \times m$ 阶矩阵 H^\dagger 是 $m \times n$ 阶矩阵 H 的广义逆矩阵,则满足

$$HH^\dagger H = H, H^\dagger HH^\dagger = H^\dagger \\ (HH^\dagger)^T = HH^\dagger, (H^\dagger H)T = H^\dagger H$$

由定义 1 可知,存在 H 的广义逆 H^\dagger ,使得 $v = H^\dagger \beta$ 。

算法步骤:

步骤 1 生成定义域内的训练点集 $t = [t_1, t_2, \dots, t_s]^T$,随机生成神经网络中输入权值向量 $w = [w_1, w_2, \dots, w_m]^T$ 以及隐藏层偏置向量 $b = [b_1, b_2, \dots, b_m]^T$,其中 m 是隐藏层节点的个数。

步骤 2 计算系数矩阵:

$$A(:, j) = w_j^n [\varphi^{(n)}(w_j t + b_j) + a(t) \varphi^{(n)}(w_j p_n t + b_j)] + \sum_{k=0}^{n-1} b_k(t) w_j^k \varphi^{(k)}(w_j p_k t + b_j) + c(t) \varphi(w_j t + b_j) \\ j = 1, 2, \dots, m$$

步骤 3 计算初值条件:

$$A_k(:, j) = w_j^k \varphi^{(k)}(w_j t_0 + b_j)$$

步骤 4 计算右端向量 $\tilde{\beta} = f(t)$ 。

步骤 5 令 $H = [A, A_1, \dots, A_k]^T$, $\beta = [\tilde{\beta}, \lambda_1, \dots, \lambda_k]^T$,并计算 H^\dagger 。

步骤 6 由 $v = H^\dagger \beta$ 得到 $v = [v_1, v_2, \dots, v_m]^T$ 。

步骤 7 将 v 代入 $N = v^T \varphi(wt + b)$ 得到比例延迟微分方程的近似解。

4 数值实验与仿真分析

在本节中,首先给出 3 个数值例子与已有方法的数值结果进行比较,说明该方法的适用性、准确性和收敛速度,再将该方法推广到双比例延迟微分系统,验证该方法在双比例延迟微分系统中也能取得较好的效果。

绝对误差定义为 $e = |N(t) - y(t)|$,其中 $N(t)$ 是方程数值解, $y(t)$ 是方程的精确解。均方根误差 (F_{RMSE}):

$$F_{RMSE} = \sqrt{\frac{\sum_{i=1}^s (N(t_i) - y(t_i))^2}{s}}$$

收敛速度 (V_{ROC}):

$$V_{ROC} = \frac{\log \frac{E_{s_1}}{E_{s_2}}}{\log \frac{s_2}{s_1}}$$

其中, s_1 和 s_2 是实例中使用的训练点个数, E_{s_1} 表示使用 s_1 个训练点得到的解中误差的均方根值, E_{s_2} 表示训练 s_2 个训练点得到的解中误差的均方根值。

4.1 一阶比例延迟微分方程

考虑一阶比例延迟微分方程:

$$\begin{cases} y'(t) = -y(t) + \frac{1}{2}y(t/2) + \frac{1}{2}y'(t/2), 0 < t < 1 \\ y(0) = 1 \end{cases}$$

该方程的精确解是 $y(t) = e^{-t}$ 。表 1 和图 2 比较了 ELM 方法 ($m = 8, s = 10$) 与两级一阶龙格-库塔法^[19] (步长取 0.01) 和变分迭代法^[20] (迭代次数取 8) 的绝对误差。将训练点数量控制在 2—32 之间,同时选取隐藏层节点数量为 5、10、15、20,分别将这些数据代入算法中,获取更多的数据以了解 ELM 方法在求解比例延迟微分方程中的准确性和收敛速度。表 2 显示了使用 ELM 方法求得数值解的均方根误差和收敛速度。

由表 1 和图 2 可以清楚看出:ELM 方法得到的数值解精度明显优于两级一阶龙格-库塔法和变分迭代法。表 2 则显示了 ELM 方法得到的精度与隐藏层节点和训练点个数相关:当增加隐藏层节点和训练点个数时,均方根值迅速减小,收敛速度也更快。

表 1 绝对误差比较

| t | 龙格-库塔法 步长 $h = 0.01$ | 变分迭代法 (迭代次数 $n = 8$) | ELM 方法 ($m = 8, n = 10$) |
|-----|-------------------------|--------------------------|-------------------------------|
| 0.2 | 8.24e-04 | 7.08e-04 | 6.21e-09 |
| 0.4 | 1.35e-03 | 1.29e-03 | 6.60e-09 |
| 0.6 | 1.66e-03 | 1.76e-03 | 6.28e-09 |
| 0.8 | 1.81e-03 | 2.15e-03 | 5.77e-09 |
| 1.0 | 1.85e-03 | 2.47e-03 | 5.22e-09 |

表 2 一阶比例延迟微分方程数值解的均方根误差和收敛速度

Table 2 RMS errors and rates of convergence of numerical solutions for the first-order pantograph delay differential equation

| 训练点 s | $m=5$ | | $m=10$ | | $m=15$ | | $m=20$ | |
|---------|------------|-----------|------------|-----------|------------|-----------|------------|-----------|
| | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} |
| 2 | 1.48e-02 | - | 1.52e-02 | - | 1.96e-02 | - | 1.61e-02 | - |
| 4 | 3.70e-04 | 5.32 | 2.74e-04 | 5.79 | 2.57e-04 | 6.25 | 1.35e-04 | 6.59 |
| 8 | 2.23e-05 | 4.05 | 2.66e-08 | 13.33 | 2.03e-08 | 13.62 | 1.41e-08 | 13.22 |
| 16 | 5.11e-06 | 2.12 | 1.48e-10 | 7.48 | 1.69e-11 | 10.23 | 2.46e-12 | 12.48 |
| 32 | 3.72e-06 | 0.45 | 1.39e-11 | 3.41 | 4.40e-13 | 5.26 | 1.10e-13 | 4.48 |

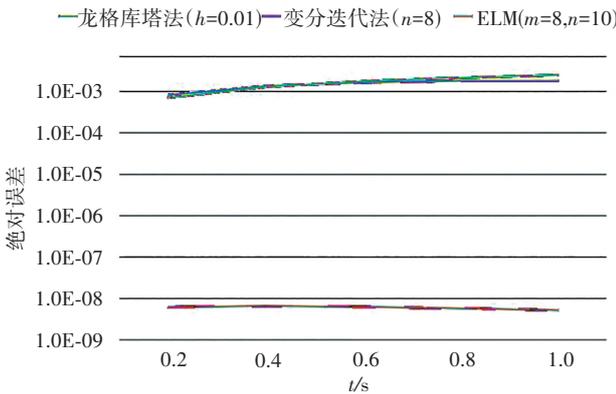


图 2 一阶比例延迟微分方程使用龙格-库塔法、变分迭代法和 ELM 方法解的绝对误差

Fig. 2 Absolute errors of the solutions using Runge-Kutta method, variational iteration method, and ELM method for the first-order pantograph delay differential equation

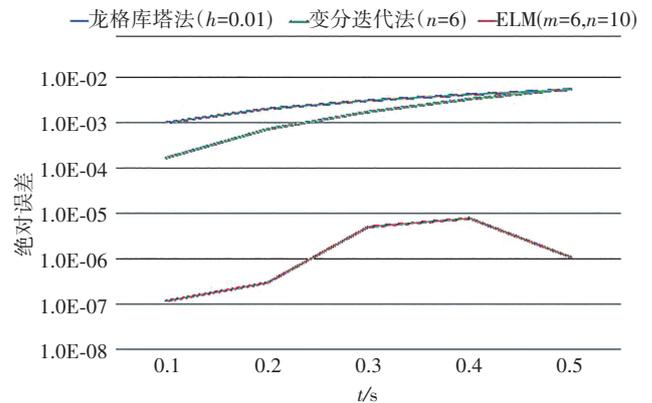


图 3 二阶比例延迟微分方程使用龙格-库塔法、变分迭代法和 ELM 方法解的绝对误差

Fig. 3 Absolute errors of the solutions using Runge-Kutta method, variational iteration method, and ELM method for the second-order pantograph delay differential equation

4.2 二阶比例延迟微分方程

考虑二阶比例延迟微分方程:

$$\begin{cases} y''(t) = \frac{3}{4}y(t) + y(t/2) + y'(t/2) + \frac{1}{2}y''(t/2) - \\ t^2 - t + 1, 0 < t < 1 \\ y(0) = y'(0) = 0 \end{cases}$$

上述方程的精确解是 $y(t) = t^2$ 。表 3 和图 3 比较了 ELM 方法 ($m=6, s=10$) 与两级一阶龙格-库塔法^[19] (步长取 0.01) 和变分迭代法^[20] (迭代次数取 6) 的绝对误差。表 4 显示了将训练点数量控制在 2—32 之间, 同时分别选取隐藏层节点数量为 5、10、15, 使用 ELM 方法求得数值解的均方根误差和收敛速度。

表 3 二阶比例延迟微分方程的绝对误差比较

Table 3 Comparison of absolute errors for the second-order pantograph delay differential equation

| t | 龙格-库塔法 步长 $h=0.01$ | 变分迭代法 (迭代次数 $n=6$) | ELM 方法 ($m=6, n=10$) |
|-----|-----------------------|------------------------|---------------------------|
| 0.1 | 1.00e-03 | 1.67e-04 | 1.16e-07 |
| 0.2 | 2.02e-03 | 7.15e-04 | 2.93e-07 |
| 0.3 | 3.07e-03 | 1.73e-03 | 5.02e-06 |
| 0.4 | 4.17e-03 | 3.30e-03 | 7.73e-06 |
| 0.5 | 5.34e-03 | 5.55e-03 | 1.08e-06 |

表 4 二阶比例延迟微分方程数值解的均方根误差和收敛速度

Table 4 RMS errors and rates of convergence of numerical solutions for the second-order pantograph delay differential equation

| 训练点 s | $m=5$ | | $m=10$ | | $m=15$ | |
|---------|------------|-----------|------------|-----------|------------|-----------|
| | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} |
| 2 | 3.45e-02 | - | 3.32e-02 | - | 3.08e-02 | - |
| 4 | 3.86e-04 | 6.48 | 2.51e-04 | 7.04 | 8.72e-05 | 8.46 |
| 8 | 1.51e-04 | 1.35 | 2.84e-07 | 9.78 | 1.03e-07 | 9.72 |
| 16 | 3.89e-05 | 1.95 | 2.35e-07 | 0.27 | 9.28e-08 | 0.15 |
| 32 | 2.34e-06 | 4.05 | 8.04e-08 | 1.54 | 3.61e-08 | 1.36 |

同 4.1 节一样, 图 3 显示了 ELM 方法在精度上明显优于其他两种方法。从表 4 中也可以看出: ELM 方法只需要少量隐藏层节点和训练点即可获得较高的精度, 且随着训练点数量和隐藏层节点数量的增加, 均方根值迅速降低, 这表明 ELM 方法确实是快速收敛的。

4.3 三阶比例延迟微分方程

考虑三阶比例延迟微分方程:

$$\begin{cases} y'''(t) = y(t) + y'(t/2) + y''(t/3) + \frac{1}{2}y'''(t/4) - \\ t^4 - \frac{1}{2}t^3 - \frac{4}{3}t^2 + 21t, 0 < t < 1 \\ y(0) = y'(0) = y''(0) = 0 \end{cases}$$

上述方程的精确解是 $y(t) = t^4$ 。表 5 和图 4 比较了 ELM 方法 ($m = 12, s = 20$) 与两级一阶龙格-库塔法^[19] (步长取 0.01) 和变分迭代法^[20] (迭代次数取 6) 的绝对误差。表 6 显示了将训练点数量控制在 4—20 之间, 同时分别选取隐藏层节点数量为 8、10、12 时, 使用 ELM 方法求得的数值解的均方根误差和收敛速度。

表 5 三阶比例延迟微分方程的绝对误差比较

Table 5 Comparison of absolute errors for the third-order pantograph delay differential equation

| t | 龙格-库塔法 步长 $h=0.01$ | 变分迭代法 (迭代次数 $n=6$) | ELM 方法 ($m=12, n=20$) |
|-----|-----------------------|------------------------|----------------------------|
| 0.2 | 4.43e-04 | 2.98e-10 | 1.24e-09 |
| 0.4 | 3.85e-03 | 1.01e-08 | 1.21e-09 |
| 0.6 | 1.39e-02 | 8.24e-08 | 1.48e-09 |
| 0.8 | 3.53e-02 | 3.76e-07 | 1.36e-09 |
| 1.0 | 7.34e-02 | 1.26e-06 | 4.65e-10 |

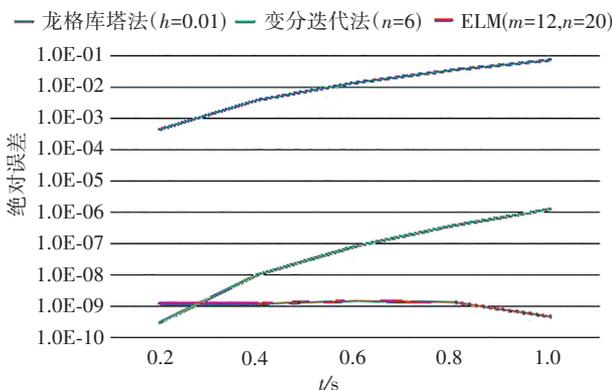


图 4 三阶比例延迟微分方程使用龙格-库塔法、变分迭代法和 ELM 方法解的绝对误差

Fig. 4 Absolute errors of the solutions using Runge-Kutta method, variational iteration method, and ELM method for the third-order pantograph delay differential equation

表 6 三阶比例延迟微分方程数值解的均方根误差和收敛速度

Table 6 RMS errors and rates of convergence of numerical solutions for the third-order pantograph delay differential equation

| 训练点 s | $m=8$ | | $m=10$ | | $m=12$ | |
|------------|------------|-----------|------------|-----------|------------|-----------|
| | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} |
| 4 | 2.66e-02 | - | 4.07e-02 | - | 1.17e-02 | - |
| 8 | 3.62e-03 | 2.87 | 1.20e-05 | 11.72 | 2.28e-06 | 12.32 |
| 12 | 5.90e-04 | 4.47 | 1.25e-06 | 5.57 | 1.82e-08 | 11.91 |
| 16 | 3.28e-05 | 10.04 | 3.20e-07 | 4.73 | 4.45e-09 | 4.89 |
| 20 | 1.47e-05 | 3.59 | 5.53e-07 | -2.45 | 2.00e-10 | 13.90 |

由以上实例可知:使用 ELM 方法得到数值解相比于两级一阶龙格-库塔法和变分迭代方法得到的解准确性更高。ELM 方法不仅只需要少量的训练点和隐藏层节点即可获得较高的精度,且随着训练点个数以及隐藏层节点数的增多,能得到更高的精度。同时 ELM 方法收敛迅速,训练过程中不需要进行迭代优化参数,不仅避免了大量的计算,而且也极大地降低了运算时间。

该方法不仅适用于求解广义比例延迟微分方程,同时适用于求解双比例延迟微分系统。由于双比例延迟微分系统中含有两个因变量,所以在使用神经网络方法求解双比例延迟微分系统需要对每一个比例延迟微分方程构建一个网络结构,每个网络对应一个因变量。

4.4 双比例延迟微分系统

考虑双比例延迟微分系统:

$$\begin{cases} y_1'(t) = y_1(t) - y_2(t) + y_1(t/2) - e^{t/2} + e^{-t} \\ y_2'(t) = -y_1(t) - y_2(t) - y_2(t/2) + e^{t/2} + e^t, 0 \leq t \leq 1 \\ y_1(0) = 1, y_2(0) = 1 \end{cases}$$

其精确解为 $y_1(t) = e^t, y_2(t) = e^{-t}$ 。表 7 和表 8 展示了训练点数量控制在 2—32 之间,同时分别选取隐藏层节点数量为 8、10、12 时,使用 ELM 方法求得的数值解的均方根误差和收敛速度。

由表 7 和表 8 可以看出:ELM 算法在求解双比例延迟微分系统中也取得了不错的效果。

表 7 双比例延迟微分系统中 $y_1(t)$ 的数值解均方根误差和收敛速度

Table 7 RMS errors and rates of convergence of numerical solutions for the system of pantograph equations with two delays $y_1(t)$

| 训练点 s | $m=8$ | | $m=10$ | | $m=12$ | |
|------------|------------|-----------|------------|-----------|------------|-----------|
| | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} |
| 2 | 3.00e-02 | - | 4.43e-02 | - | 6.05e-02 | - |
| 4 | 6.72e-04 | 5.48 | 1.15e-03 | 5.26 | 7.93e-04 | 6.25 |
| 8 | 2.44e-06 | 8.10 | 2.40e-08 | 15.54 | 1.08e-07 | 12.84 |
| 16 | 4.88e-07 | 2.32 | 9.87e-09 | 1.28 | 3.98e-10 | 8.08 |
| 32 | 3.65e-08 | 3.74 | 2.56e-10 | 5.26 | 7.64e-11 | 2.38 |

表 8 双比例延迟微分系统 $y_2(t)$ 的数值解的均方根误差和收敛速度

Table 8 RMS errors and rates of convergence of numerical solutions for the system of pantograph equations with two delays $y_2(t)$

| 训练点 s | $m=8$ | | $m=10$ | | $m=12$ | |
|------------|------------|-----------|------------|-----------|------------|-----------|
| | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} | F_{RMSE} | V_{ROC} |
| 2 | 2.11e-03 | - | 3.72e-03 | - | 5.59e-03 | - |
| 4 | 4.31e-04 | 2.29 | 1.40e-04 | 4.73 | 9.11e-05 | 5.93 |
| 8 | 3.46e-07 | 10.28 | 6.61e-09 | 14.37 | 2.43e-09 | 15.19 |
| 16 | 3.44e-08 | 3.33 | 5.96e-10 | 3.47 | 2.46e-12 | 9.94 |
| 32 | 4.03e-09 | 3.09 | 6.60e-11 | 3.17 | 3.61e-12 | -0.55 |

5 结果与讨论

本文使用基于极限学习机算法的前馈神经网络方法求解比例延迟微分方程及双比例延迟微分系统。ELM 算法完全消除了迭代算法的优化过程,使得计算复杂度远低于其他传统数值方法。该方法不仅只需要少量的训练点和隐藏层节点即可获得较高的精度,且随着训练点个数以及隐藏层节点数的增多,能得到更高的精度。数值实验结果表明:该方法比以往的一些方法精度更高,收敛速度更快。因此,极限学习机算法是求解比例延迟微分方程和双比例延迟微分系统数值解强有力的数学工具。

由第三节所呈现的图表可以看出:ELM 方法在所有情况下的收敛速度都不是一致的。缺乏统一的收敛速度是因为在训练点数量和隐藏层节点数量相同的情况下,由于随机生成的输入权值和隐藏层偏置每一次都不同,所以每次都会得到不同的结果。在进一步的工作中,可以考虑对神经网络参数进行预处理,以便可以在不同问题中获得一致的收敛速度。

参考文献(References):

- [1] YANG H, WU F, KLOEDEN P E. Stationary distribution of stochastic population dynamics with infinite delay[J]. *Journal of Differential Equations*, 2022, 340: 205—226.
- [2] RIHAN F A, ALSAKAJI H J, KUNDU S, et al. Dynamics of a time-delay differential model for tumour-immune interactions with random noise[J]. *Alexandria Engineering Journal*, 2022, 61(12): 11913—11923.
- [3] AL-DARABSAH I. Threshold dynamics of a time-delayed epidemic model for continuous imperfect-vaccine with a generalized nonmonotone incidence rate[J]. *Nonlinear Dynamics*, 2020, 101(2): 1281—1300.
- [4] JI X A, OROSZ G. Learning time delay systems with neural ordinary differential equations [J]. *IFAC-Papers Online*, 2022, 55(36): 79—84.
- [5] BARTFAI A, DOMBOVARI Z. Hopf bifurcation calculation in neutral delay differential equations: Nonlinear robotic arms subject to delayed acceleration feedback control [J]. *International Journal of Non-Linear Mechanics*, 2022, 147: 104239—104249.
- [6] OCKENDON J R, TAYLER A B. The dynamics of a current collection system for an electric locomotive[J]. *Proceedings of the Royal Society A: Mathematical*, 1971, 322(1551): 447—468.
- [7] WANG W. High order stable Runge-Kutta methods for nonlinear generalized pantograph equations on the geometric mesh[J]. *Applied Mathematical Modelling*, 2015, 39(1): 270—283.
- [8] YUZBASI S, SEZER M. Shifted Legendre approximation with the residual correction to solve pantograph-delay type differential equations [J]. *Applied Mathematical Modelling*, 2015, 39(21): 6529—6542.
- [9] DOHA E H, BHRAWY A H, HAFEZ R M. Numerical algorithm for solving multi-pantograph delay equations on the half-line using Jacobi rational functions with convergence analysis [J]. *Acta Mathematicae Applicatae Sinica*, 2017, 33(2): 297—310.
- [10] CAKMAK M, ALKAN S. A numerical method for solving a class of systems of nonlinear pantograph differential equations [J]. *Alexandria Engineering Journal*, 2022, 61(4): 2651—2661.
- [11] BAHGAT M. Approximate analytical solution of the linear and nonlinear multi-pantograph delay differential equations [J]. *Physica Scripta*, 2020, 95(5): 055219—055229.
- [12] MALL S, CHAKRAVERTY S. Single layer Chebyshev neural network model for solving elliptic partial differential equations[J]. *Neural Processing Letters*, 2017, 45(3): 1—16.
- [13] SUN H, HOU M, YANG Y, et al. Solving partial differential equation based on Bernstein neural network and extreme learning machine algorithm [J]. *Neural Processing Letters*, 2019, 50: 1153—1172.
- [14] RAJA M. Numerical treatment for boundary value problems of pantograph functional differential equation using computational intelligence algorithms [J]. *Applied Soft Computing*, 2014, 24: 806—821.
- [15] KHAN I, RAJA M, SHOAIB M, et al. Design of neural network with Levenberg-Marquardt and Bayesian regularization backpropagation for solving pantograph delay differential equations[J]. *IEEE Access*, 2020(8): 137918—137933.
- [16] SABIR Z, RAJA M, GUIRAO J, et al. Meyer wavelet neural networks to solve a novel design of fractional order pantograph Lane-Emden differential model [J]. *Chaos, Solitons & Fractals*, 2021, 152: 111404—111414.
- [17] PANGHAL S, KUMAR M. Neural network method: Delay and system of delay differential equations [J]. *Engineering with Computers*, 2021, 38(2): 2423—2432.
- [18] HUANG G B, ZHU Q Y, SIEW C K. Extreme learning machine: Theory and applications [J]. *Neurocomputing*, 2006, 70(1-3): 489—501.
- [19] BELLEN A, ZENNARO M. Numerical methods for delay differential equations [M]. New York: The Clarendon Press Oxford University Press, 2003.
- [20] CHEN X, WANG L. The variational iteration method for solving a neutral functional-differential equation with proportional delays [J]. *Computers & Mathematics with Applications*, 2010, 59(8): 2696—2702.