利用神经元拓展正则极端学习机预测时间序列

张 弦 王宏力

(第二炮兵工程学院 自动控制工程系, 西安 710025)

摘 要: 为实现对于时间序列预测数据的准确预测,提出一种神经元拓展正则极端学习机(NERELM,Neuron-Expanding Regularized Extreme Learning Machine),并研究了其在时间序列预测中的应用. NERELM 根据结构风险最小化原理权衡经验风险与结构风险,以逐次拓展隐层神经元的方式自动确定最佳的网络结构,以避免传统神经网络训练过程中需人为确定网络结构的弊端. 应用于时间序列的仿真结果表明:NERELM 可有效实现对于 RELM 最佳网络结构的自动确定,具有预测精度高与计算速度快的优点.

关 键 词: 神经网络; 极端学习机; 正则极端学习机; 时间序列预测

中图分类号: TP 277

文献标识码: A **文 章 编 号**: 1001-5965(2011)12-1510-05

Time series prediction using neuron-expanding regularized extreme learning machine

Zhang Xian Wang Hongli

(Department of Automatic Control Engineering, The Second Artillery Engineering College, Xi'an 710025 China)

Abstract: A new algorithm called neuron-expanding regularized extreme learning machine (NERELM) was proposed and applied to time series prediction. In order to enhance the generalization performance of NERELM, the empirical risk and the structural risk were balanced on the basis of structural risk minimization theory. The output weights of NERELM were analytically determined at extremely fast learning speed instead of using gradient-based learning algorithm, and the optimal network structure of NERELM was automatically determined by expanding its hidden neuron nodes iteratively. Experiments on time series prediction indicate that NERELM has better performance in training computation cost and prediction accuracy in comparison with conventional gradient-based neural networks.

Key words: neural networks; extreme learning machine; regularized extreme learning machine; time series prediction

时间序列是指按照出现时间的先后次序排列的某一变量或指标的观测值,反映了现象与现象之间关系的发展变化规律,因而包含了系统结构特征及其运行规律,所以可通过对时间序列的预测来揭示所研究系统的未来行为[1-4].

神经网络是一种由大量神经元互连而成的复杂非线性系统,具有较强的自适应性、自学习能力与非线性映射能力,已被许多学者用于解决工程领域中的实际问题[5-8]. 然而,传统的神经网络的训练过程需根据训练目标与梯度信息对网络权值进行反复修改,整个过程通常需几百甚至几千次迭代才可最终完成,计算量较大.

极端学习机[9](ELM, Extreme Learning Machine)是一种新颖的单隐层前馈神经网络. ELM 在随机给定输入权值与神经元参数的基础上,将传统前馈神经网络参数(包括网络权值与神经元参数)训练问题转化为求解线性方程组,以直接计算输出权值的最小二乘解的方式完成网络训练

收稿日期: 2010-07-23; **网络出版时间**: 2011-11-14 17:19; **DOI**: CNKI:11-2625/V.20111114.1719.010
网络出版地址: www.cnki.net/kcms/detail/11.2625.V.20111114.1719.010.html
作者简介: 张 弦(1982-),男,湖北武汉人,博士生,sltecas@163.com.

过程. 与传统单隐层前馈神经网络训练算法中的网络权值迭代调整策略相比,ELM 的训练速度显著提升. 这一速度优势使得 ELM 已在工程领域获得了成功应用[10-19]. 为改善 ELM 的泛化能力,文献[16]借鉴统计学习理论中的结构风险最小化原理,提出了正则极端学习机(RELM ,Regularized Extreme Learning Machine). RELM 通过引入调节参数权衡 ELM 的结构风险与经验风险,有效提高了 ELM 的泛化能力. 研究表明[9]:隐层神经元的数量决定了 ELM 的学习精度与泛化能力,同时也是在设计 ELM 网络结构时必须事先确定的关键因素. 在 RELM 的网络结构设计时,也存在相同的问题. 隐层神经元过少会导致 RELM 逼近任意非线性系统的能力有限,丧失部分必要的学习精度;隐层神经元过多虽可保证 RELM 的学习精度,但会导致此时 RELM 的网络结构过大,存在较多冗余的隐层神经元,从而增加了训练时间.

针对应用于时间序列预测问题中的 RELM 网络结构设计问题,本文研究并提出一种可自动确定最佳隐层神经元数量的神经元拓展正则极端学习机(NERELM, Neuron-Expanding Regularized Extreme Learning Machine),其有效性将通过仿真时间序列预测实例加以验证.

1　正则极端学习机

假设包含 M 个数据的时间序列数据集为 $\{x_i\}_{i=1}^{M}$,其中 x_i 为时间序列数据. 在嵌入维数为 n 的条件下将 $\{x_i\}_{i=1}^{M}$ 转化为训练样本集 $S=\{(\boldsymbol{x}_i, t_i)\}_{i=1}^{N}$,其中 $\boldsymbol{x}_i=[x_i \quad x_{i+1} \quad \cdots \quad x_{i+n-1}]^{\mathrm{T}}$ 为输入,$t_i=x_{i+n}$ 为输出,$N=M-n$ 为训练样本数量. 根据结构风险最小化原理,为较好权衡结构风险与经验风险,RELM 通过引入参数 γ 调节两种风险的比例. 一个以 $S=\{(\boldsymbol{x}_i,t_i)\}_{i=1}^{N}$ 为训练样本集,且包含 L 个隐藏层神经元函数 $f(\cdot)$ 的 RELM 回归问题可表示如下式所示的约束优化:

$$\left.\begin{aligned} J &= \frac{1}{2}\boldsymbol{\beta}_L^{\mathrm{T}}\boldsymbol{\beta}_L + \frac{\gamma}{2}\sum_{i=1}^{N}\varepsilon_i^2 \\ \text{s.t.}\quad t_i &= \sum_{k=1}^{L}\beta_k f(\boldsymbol{\alpha}_k\boldsymbol{x}_i + b_k) - \varepsilon_i \\ & i = 1,2,\cdots,N \end{aligned}\right\} \tag{1}$$

式中,$f(\boldsymbol{\alpha}_k\boldsymbol{x}_i+b_k)=1/\exp[-(\boldsymbol{\alpha}_k\boldsymbol{x}_i+b_k)]$;$\boldsymbol{\alpha}_k=[\alpha_{k1} \quad \alpha_{k2} \quad \cdots \quad \alpha_{kn}]$ 为连接第 k 个隐层神经元的输入权值;$\boldsymbol{\beta}_L=[\beta_1 \quad \beta_2 \quad \cdots \quad \beta_L]^{\mathrm{T}}$;$\beta_k$ 为连接第 k 个隐层神经元的输出权值;b_k 为第 k 个隐层神经元的偏差;ε_i 为回归误差. 为求解式(1)所示的优化问题, 建立相应的拉格朗日函数为

$$L(\boldsymbol{w}_L,\boldsymbol{\varepsilon}_L,\boldsymbol{\beta}_L) = \frac{1}{2}\boldsymbol{\beta}_L^{\mathrm{T}}\boldsymbol{\beta}_L + \frac{\gamma}{2}\boldsymbol{\varepsilon}_L^{\mathrm{T}}\boldsymbol{\varepsilon}_L - \boldsymbol{w}_L(\boldsymbol{H}_L\boldsymbol{\beta}_L - \boldsymbol{T}_L - \boldsymbol{\varepsilon}_L) \tag{2}$$

式中,$\boldsymbol{w}_L=[w_1 \quad w_2 \quad \cdots \quad w_N]$ 为拉格朗日权值;

$$\boldsymbol{H}_L = \begin{bmatrix} f(\boldsymbol{\alpha}_1\boldsymbol{x}_1+b_1) & f(\boldsymbol{\alpha}_2\boldsymbol{x}_1+b_2) & \cdots & f(\boldsymbol{\alpha}_L\boldsymbol{x}_1+b_L) \\ f(\boldsymbol{\alpha}_1\boldsymbol{x}_2+b_1) & f(\boldsymbol{\alpha}_2\boldsymbol{x}_2+b_2) & \cdots & f(\boldsymbol{\alpha}_L\boldsymbol{x}_2+b_L) \\ \vdots & \vdots & \ddots & \vdots \\ f(\boldsymbol{\alpha}_1\boldsymbol{x}_N+b_1) & f(\boldsymbol{\alpha}_2\boldsymbol{x}_N+b_2) & \cdots & f(\boldsymbol{\alpha}_L\boldsymbol{x}_N+b_L) \end{bmatrix}$$

为神经元矩阵;$\boldsymbol{\varepsilon}_L=[\varepsilon_1 \quad \varepsilon_2 \quad \cdots \quad \varepsilon_N]^{\mathrm{T}}$;$\boldsymbol{T}_L=[t_1 \quad t_2 \quad \cdots \quad t_N]^{\mathrm{T}}$.

对式(2)中的 $L(\boldsymbol{w}_L,\boldsymbol{\varepsilon}_L,\boldsymbol{\beta}_L)$ 中的各变量求偏导数,并令偏导数为 0 可得

$$\left.\begin{aligned} &\frac{\partial L}{\partial \boldsymbol{\beta}_L} \rightarrow \boldsymbol{\beta}_L^{\mathrm{T}} = \boldsymbol{w}_L\boldsymbol{H}_L \\ &\frac{\partial L}{\partial \boldsymbol{\varepsilon}_L} \rightarrow \gamma\boldsymbol{\varepsilon}_L^{\mathrm{T}} + \boldsymbol{w}_L = 0 \\ &\frac{\partial L}{\partial \boldsymbol{w}_L} \rightarrow \boldsymbol{H}_L\boldsymbol{\beta}_L - \boldsymbol{T}_L - \boldsymbol{\varepsilon}_L = 0 \end{aligned}\right\} \tag{3}$$

联立式(3)可得 $\boldsymbol{\beta}_L$ 的解为

$$\tilde{\boldsymbol{\beta}}_L = (\boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{H}_L + \boldsymbol{I}_L/\gamma)^{-1}\boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{T}_L \tag{4}$$

式中,$\boldsymbol{\beta}_L=[\tilde{\beta}_1 \quad \tilde{\beta}_2 \quad \cdots \quad \tilde{\beta}_L]^{\mathrm{T}}$;$\boldsymbol{I}_L\in\mathbf{R}^{L\times L}$ 为单位矩阵. 在求解出 $\tilde{\boldsymbol{\beta}}_L$ 的基础上,则可利用 $\tilde{\boldsymbol{\beta}}_L$ 建立训练后的时间序列预测模型为

$$v = \sum_{k=1}^{L}\tilde{\beta}_k f(\boldsymbol{\alpha}_k\boldsymbol{u} + b_k) \tag{5}$$

式中,$\boldsymbol{u}\in\mathbf{R}^n$,$v\in\mathbf{R}$ 分别为预测模型输入、输出.

2　输出权值解的递推求解方法

假设为提高式(5)所示的 RELM 时间序列预测模型的学习精度,需将其隐层神经元数量从 L 个增加为 $L+\Delta L$ 个(ΔL 为 L 的增加梯度). 从式(1)~式(4)所示的 RELM 训练过程可知:当 L 发生变化时,输出权值 $\boldsymbol{\beta}_L$ 的解 $\tilde{\boldsymbol{\beta}}_L$ 也将随之变化,需重新计算隐层神经元数量为 $L+\Delta L$ 个时的输出权值 $\boldsymbol{\beta}_{L+\Delta L}$ 的解 $\tilde{\boldsymbol{\beta}}_{L+\Delta L}$.

为有效解决上述问题,本文提出一种在 L 递增条件下,RELM 输出权值解的递推求解方法,使得 $\tilde{\boldsymbol{\beta}}_{L+\Delta L}$ 可在 $\tilde{\boldsymbol{\beta}}_L$ 的基础上递推而得,从而避免输出权值解的重新计算. 当 L 增加为 $L+\Delta L$ 个时,$\boldsymbol{H}_L$ 将变化为

$$\boldsymbol{H}_{L+\Delta L} = [\boldsymbol{H}_L \quad \boldsymbol{h}_{L+1} \quad \boldsymbol{h}_{L+2} \quad \cdots \quad \boldsymbol{h}_{L+\Delta L}] \tag{6}$$

式中,$\boldsymbol{h}_{L+i}=[f(\boldsymbol{\alpha}_{L+i}\boldsymbol{x}_1+b_{L+i}) \quad f(\boldsymbol{\alpha}_{L+i}\boldsymbol{x}_2+b_{L+i}) \quad \cdots \quad f(\boldsymbol{\alpha}_{L+i}\boldsymbol{x}_N+b_{L+i})]^{\mathrm{T}}$,$i=1,2,\cdots,\Delta L$. 因此有

$$\boldsymbol{H}_{L+\Delta L}^{\mathrm{T}}\boldsymbol{H}_{L+\Delta L}+\frac{\boldsymbol{I}_{L+\Delta L}}{\gamma}=$$

$$\begin{bmatrix}\boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{H}_L+\dfrac{\boldsymbol{I}_L}{\gamma} & \boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{h}_{L+1} & \cdots & \boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{h}_{L+\Delta L}\\ \boldsymbol{h}_{L+1}^{\mathrm{T}}\boldsymbol{H}_L & \boldsymbol{h}_{L+1}^{\mathrm{T}}\boldsymbol{h}_{L+1}+\dfrac{1}{\gamma} & \cdots & \boldsymbol{h}_{L+1}^{\mathrm{T}}\boldsymbol{h}_{L+\Delta L}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{h}_{L+\Delta L}^{\mathrm{T}}\boldsymbol{H}_L & \boldsymbol{h}_{L+\Delta L}^{\mathrm{T}}\boldsymbol{h}_{L+1} & \cdots & \boldsymbol{h}_{L+\Delta L}^{\mathrm{T}}\boldsymbol{h}_{L+\Delta L}+\dfrac{1}{\gamma}\end{bmatrix}=$$

$$\begin{bmatrix}\boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{H}_L+\dfrac{\boldsymbol{I}_L}{\gamma} & \boldsymbol{Q}_{L+\Delta L}\\ \boldsymbol{Q}_{L+\Delta L}^{\mathrm{T}} & \boldsymbol{G}_{L+\Delta L}\end{bmatrix} \tag{7}$$

式中，$\boldsymbol{I}_{L+\Delta L}\in\mathbf{R}^{(L+\Delta l)\times(L+\Delta L)}$ 为单位矩阵；

$$\boldsymbol{Q}_{L+\Delta L}=[\boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{h}_{L+1}\quad \boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{h}_{L+2}\quad \cdots\quad \boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{h}_{L+\Delta L}]$$

令 $\boldsymbol{S}_{L+\Delta L}=\boldsymbol{H}_{L+\Delta L}^{\mathrm{T}}\boldsymbol{H}_{L+\Delta L}+\boldsymbol{I}_{L+\Delta L}/\gamma$，$\boldsymbol{S}_L=\boldsymbol{H}_L^{\mathrm{T}}\boldsymbol{H}_L+\boldsymbol{I}_L/\gamma$，将分块矩阵求逆引理[20]应用于式(7)可得

$$\boldsymbol{S}_{L+\Delta L}^{-1}=\begin{bmatrix}\boldsymbol{S}_L & \boldsymbol{Q}_{L+\Delta L}\\ \boldsymbol{Q}_{L+\Delta L}^{\mathrm{T}} & \boldsymbol{G}_{L+\Delta L}\end{bmatrix}^{-1}=$$

$$\begin{bmatrix}\boldsymbol{S}_L^{-1}+\boldsymbol{S}_L^{-1}\boldsymbol{Q}_{L+\Delta L}\boldsymbol{P}_{L+\Delta L}\boldsymbol{Q}_{L+\Delta L}^{\mathrm{T}}\boldsymbol{S}_L^{-1} & -\boldsymbol{S}_L^{-1}\boldsymbol{Q}_{L+\Delta L}\boldsymbol{P}_{L+\Delta L}\\ -\boldsymbol{P}_{L+\Delta L}\boldsymbol{Q}_{L+\Delta L}^{\mathrm{T}}\boldsymbol{S}_L^{-1} & \boldsymbol{P}_{L+\Delta L}\end{bmatrix} \tag{8}$$

式中，$\boldsymbol{P}_{L+\Delta L}=[\boldsymbol{G}_{L+\Delta L}-\boldsymbol{Q}_{L+\Delta L}^{\mathrm{T}}\boldsymbol{S}_L^{-1}\boldsymbol{Q}_{L+\Delta L}]^{-1}$. 因此 $\boldsymbol{S}_{L+\Delta L}^{-1}$ 可在 $\boldsymbol{S}_L^{-1}$ 的基础上根据式(8)递推而得，且计算 $\Delta L\times\Delta L$ 维矩阵 $\boldsymbol{P}_{L+\Delta L}$ 逆的代价远小于直接对 $(L+\Delta L)\times(L+\Delta L)$ 维矩阵 $\boldsymbol{S}_{L+\Delta L}$ 进行逆运算，因此可加快 $\tilde{\boldsymbol{\beta}}_{L+\Delta L}$ 的求解速度.

3 神经元拓展正则极端学习机

将 RELM 应用于时间序列预测时，当时间序列的嵌入维数确定之后，为获得式(5)所示的时间序列预测模型，L 是必须事先确定的关键因素. 然而由于实际应用于 RELM 预测模型训练的时间序列训练样本的复杂性，因此难以准确地给出最佳的隐层神经元数量，使得具有该数量神经元的 RELM 预测模型对于训练样本的学习精度以及对于测试样本的泛化能力均为理想，同时还可保持简约的模型结构. 为解决在 RELM 训练过程中 L 的有效选取的问题，本文提出可自动确定最佳隐层神经元数量的 NERELM，L 采取逐次拓展的方式自动确定，其训练过程为如下：

1) 令 RELM 的 $L=1$，并将训练样本集 $S=\{(\boldsymbol{x}_i,t_i)\}_{i=1}^{N}$ 平均分割为两个不重叠的子集 $S_1=\{(\boldsymbol{x}_i,t_i)\}_{i=1}^{N/2}$ 与 $S_2=\{(\boldsymbol{x}_i,t_i)\}_{i=N/2+1}^{N}$；

2) 利用 $S_1=\{(\boldsymbol{x}_i,t_i)\}_{i=1}^{N/2}$ 训练隐层神经元数量为 L 的 RELM 预测模型，根据式(4)计算与之对应的 $\boldsymbol{\beta}_L$ 的解 $\tilde{\boldsymbol{\beta}}_L$，利用 $\tilde{\boldsymbol{\beta}}_L$ 建立式(5)所示的具有 L 个隐层神经元的 RELM 预测模型，并用 $S_2=\{(\boldsymbol{x}_i,t_i)\}_{i=N/2+1}^{N}$ 测试训练后的 RELM 预测模型；

3) 计算此时 RELM 预测模型对于 $S_1=\{(\boldsymbol{x}_i,t_i)\}_{i=1}^{N/2}$ 的训练误差与对于 $S_2=\{(\boldsymbol{x}_i,t_i)\}_{i=N/2+1}^{N}$ 的测试误差之和为

$$C_L=\sum_{i=1}^{N/2}|E_{1i}|+\sum_{i=1}^{N/2}|E_{2i}| \tag{9}$$

式中，E_{1i} 为 RELM 预测模型的训练误差；E_{2i} 为 RELM 预测模型的测试误差；

4) 令 $L=L+\Delta L$ 后转至 2)，从 $L=2$ 开始判断是否同时满足下式所示的不等式关系：

$$|C_L-C_{L-1}|\leqslant\theta \tag{10}$$

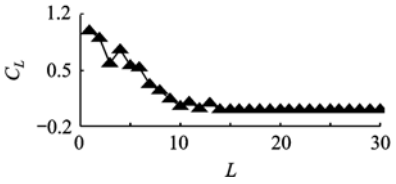
式中，θ 为预先设定的学习精度. 如果满足，则 RELM 训练过程结束，以 L 为最佳的 RELM 隐层神经元数量，并建立 RELM 预测模型，否则继续增加 L 直至满足条件式(10)所示的条件.

4 时间序列预测

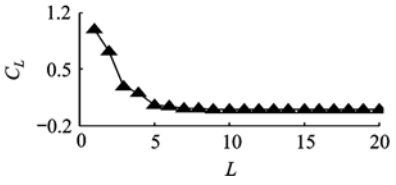
以 6 种典型时间序列为例，包括 3 种混沌时间序列：逻辑斯提克(Logistic)、洛伦兹(Lorenz)与麦克-格拉斯(Mackey-Glass)，以及 3 种来自时间序列预测小组[21]的时间序列：太阳黑子(Sunspot)、圣达菲激光(Santa Fe laser)与波兰电力(Poland electricity). 每种时间序列包含 500 个数据，选择 Logistic，Lorenz 与 Mackey-Glass 的嵌入维数为 6，Sunspot，Santa Fe laser 与 Poland electricity 的嵌入维数为 10. 设定 $\gamma=10^8$，$\theta=10^{-2}$，利用前 200 个数据训练 RELM，并对后 300 个数据进行预测验证，以预测平均相对误差 E_L 衡量预测精度. 图 1 展示了 C_L 随 L 的变化规律(图中所示的 C_L 已被归一化至[0,1]区间内.).

从图 1 可见，C_L 随 L 的增加而表现出逐渐减小的变化趋势，表明通过增加 L 可有效提高 RELM 对于训练样本的学习精度与对于测试样本的泛化能力. 同时从图 1 还可发现，C_L 的减小幅度会随着 L 的逐渐增加而逐渐减小，直至无明显变化，例如，图 1a 中当 $L\geqslant20$ 时，图 1c 中当 $L\geqslant10$ 时与图 1e 中当 $L\geqslant30$ 时. 这说明此时的 RELM 的学习精度与泛化能力已达到最佳值，继续增加 L 已无法对其进行进一步改善. 因此，在 C_L 停止减小时停止对 RELM 神经元的增长，则可使此时的 RELM 具有最佳的隐层神经元数量.

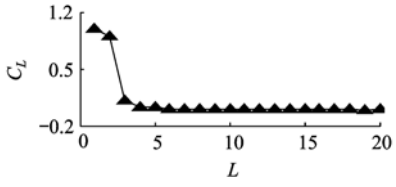
由于 C_L 综合反映了 RELM 的学习与泛化能



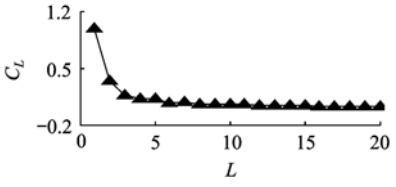
a　Logistic



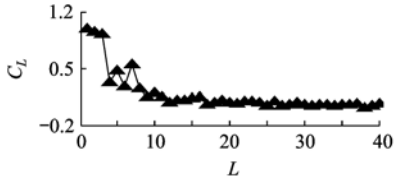
b　Lorenz



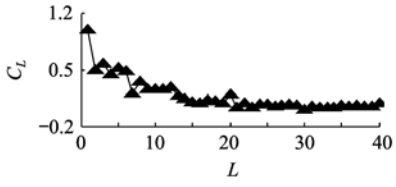
c　Mackey-Glass



d　Sunspot



e　Santa Fe laser



f　Poland electricity

图 1　在 L 递增条件下 C_L 的变化规律

力,因此,根据 C_L 的变化规律所确定的 RELM 最佳隐层神经元数量,可使基于此的 RELM 预测模型具有理想的学习与泛化能力,提高其预测精度.

表 1 为 NERELM 对于 6 种时间序列预测的最佳隐层神经元数量以及预测平均相对误差.

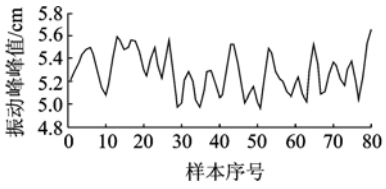
表 1　NERELM 对于 6 种时间序列的预测结果

时间序列	最佳隐层神经元数量	平均相对误差/%
Logistic	19	1.189 1
Lorenz	13	1.381 3
Mackey-Glass	9	0.469 9
Sunspot	14	5.509 8
Santa Fe laser	33	4.235 1
Poland electricity	38	7.564 7

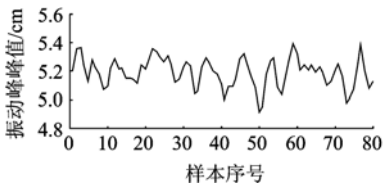
从表 1 中数据可见,具有利用本文方法所确定的最佳隐层神经元数量的 RELM 对于 6 种时间序列的预测平均相对误差均较小,因此实现了对于时间序列数据的准确预测.

5　工程应用

以两组液压泵振动数据为研究对象,其中一组为端盖振动信号,另一组为壳体振动信号. 选取其中的 80 个端盖振动峰峰值与 80 个壳体振动峰峰值进行建模预测,这两组时间序列如图 2 所示.



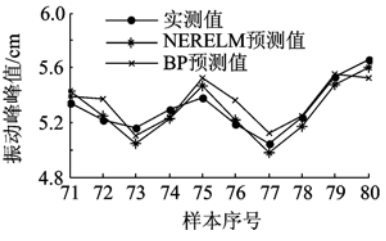
a　端盖



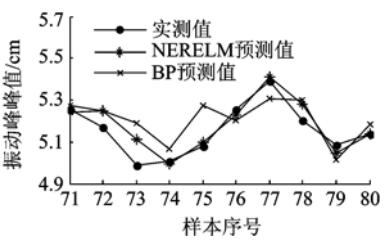
b　壳体

图 2　液压泵端盖与壳体振动峰峰值

选择嵌入维数为 6,设定 $\gamma = 10^8$,$\theta = 10^{-2}$,利用前 70 个数据训练 NERELM 预测模型,对后 10 个数据进行预测验证,同时与经典反向传播(BP, Backpropagation)神经网络比较. 经神经元拓展学习,用于端盖与壳体振动峰峰值预测的 NERELM 的 L 为 26 与 33,BP 神经网络的 L 设定与 NERELM 相同. 图 3 与表 2 为预测结果,表 2 中的预测时间为从模型训练到完成预测用的总时间.



a　端盖



b　壳体

图 3　液压泵端盖与壳体的预测结果

表 2 NERELM 与 BP 神经网络的预测结果比较

液压泵	NERELM		BP 神经网络	
	平均相对误差/%	预测时间/s	平均相对误差/%	预测时间/s
端盖	0.8198	0.0079	1.7689	0.3170
壳体	1.2394	0.0106	1.6836	0.2575

RELM 通过引入参数 γ 权衡 ELM 的结构风险与经验风险,其泛化能力较 BP 神经网络有显著提高[16].作为一种可自动确定最佳网络结构的 RELM 改进模式,NERELM 同样具备了这一优势,因此相比于 BP 神经网络,NERELM 对于端盖与壳体振动峰峰值的预测平均相对误差更小,预测准确性更高. 同时 NERELM 继承了 ELM 的计算快速性,因此相比于需多次迭代才可确定网络权值的 BP 神经网络,NERELM 的平均预测时间仅为 BP 神经网络的 1/30 左右,优势极为明显. 因此综合考虑预测准确性与计算快速性,NERELM 比 BP 神经网络更适于时间序列预测.

6 结 论

不同于传统神经网络训练算法中的网络权值迭代调整策略,RELM 以求解线性方程组的新颖方式完成网络训练过程,极大提升了神经网络的训练速度,因此具有较高的学术研究与工程应用价值. 本文提出的 NERELM 集 ELM 的计算快速性与 RELM 的泛化能力强这两大优势于一身,同时还可通过神经元拓展方式自动确定最佳的网络结构,进一步减少了需人为确定的因素. 时间序列预测实例表明: 利用 NERELM 可实现对于时间序列数据的准确预测. 因此,NERELM 可为时间序列预测提供一条更为有效且准确的解决途径.

参考文献 (References)

[1] Song Q S, Feng Z R. The hybrid forecasting model based on chaotic mapping, genetic algorithm and support vector machine[J]. Expert Systems with Applications, 2010, 37(2): 1776 – 1783

[2] Jeng J T, Chuang C C, Tao C W. Hybrid SVMR-GPR for modeling of chaotic time series systems with noise and outliers[J]. Neurocomputing, 2010, 73(10 – 12): 1686 – 1693

[3] Su L Y. Prediction of multivariate chaotic time series with local polynomial fitting[J]. Computers & Mathematics with Applications, 2010, 59(2): 737 – 744

[4] Lau K W, Wu Q H. Local prediction of non-linear time series using support vector regression[J]. Pattern Recognition, 2008, 41(5): 1539 – 1547

[5] Muhammad A F, Zolfaghari S. Chaotic time series prediction with residual analysis method using hybrid Elman NARX neural networks[J]. Neurocomputing, 2010, 73(13 – 15): 2540 – 2553

[6] Song Q S, Feng Z R. Effects of connectivity structure of complex echo state network on its prediction performance for nonlinear time series[J]. Neurocomputing, 2010, 73(10 – 12), 2177 – 2185

[7] Fu Y Y, Wu C J, Jeng J T, et al. ARFNNs with SVR for prediction of chaotic time series with outliers[J]. Expert Systems with Applications, 2010, 37(6): 4441 – 4451

[8] Lin C J, Chen C H, Lin C T. A hybrid of cooperative particle swarm optimization and cultural algorithm for neural fuzzy networks and its prediction applications[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2008, 39(1): 55 – 68

[9] Huang G B, Zhu Q Y, Siew C K. Extreme learning machine: theory and applications[J]. Neurocomputing, 2006, 70(1 – 3): 489 – 501

[10] Feng G R, Huang G B, Lin Q P, et al. Error minimized extreme learning machine with growth of hidden nodes and incremental learning[J]. IEEE Transactions on Neural Networks, 2009, 20(8): 1352 – 1357

[11] Tang X L, Han M. Partial Lanczos extreme learning machine for single-output regression problems[J]. Neurocomputing, 2009, 72(13 – 15): 3066 – 3076

[12] Cao J W, Lin Z P, Huang G B. Composite function wavelet neural networks with extreme learning machine[J]. Neurocomputing, 2010, 73(7 – 9): 1405 – 1416

[13] Minhas R, Baradarani A, Seifzadeh S, et al. Human action recognition using extreme learning machine based on visual vocabularies[J]. Neurocomputing, 2010, 73(16 – 18): 1906 – 1917

[14] Huang G B, Ding X J, Zhou H M. Optimization method based extreme learning machine for classification[J]. Neurocomputing, 2010, 74(1 – 3): 155 – 163

[15] Deng W Y, Zheng Q H, Lian S G, et al. Ordinal extreme learning machine[J]. Neurocomputing, 2010, 74(1 – 3): 447 – 456

[16] 邓万宇, 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
Deng Wanyu, Zheng Qinghua, Chen Lin, et al. Research on extreme learning of neural networks[J]. Chinese Journal of Computers, 2010, 33(2): 279 – 287 (in Chinese)

[17] Deng J, Li k, Irwin G W. Fast automatic two-stage nonlinear model identification based on the extreme learning machine[J]. Neurocomputing, 2011, 74(16): 2422 – 2429

[18] Wang Y G, Cao F L, Yuan Y B. A study on effectiveness of extreme learning machine[J]. Neurocomputing, 2011, 74(16): 2483 – 2490

[19] Heeswijk M V, Miche Y, Oja E, et al. GPU-accelerated and parallelized ELM ensembles for large-scale regression[J]. Neurocomputing, 2011, 74(16): 2430 – 2437

[20] 张贤达. 矩阵分析与应用[M]. 北京: 清华大学出版社, 2005
Zhang Xianda. Matrix analysis and applications[M]. Beijing: Tsinghua University Press, 2005 (in Chinese)

[21] Time series prediction group[EB/OL]. http://www.cis.hut.fi/projects/tsp/? page = Timeseries, 2007

(编 辑: 刘登敏)