

基于模糊边界模块化神经网络的混沌时间序列预测*

马千里[†] 郑启伦 彭 宏 覃姜维

(华南理工大学计算机科学与工程学院, 广州 510640)

(2007 年 10 月 17 日收到, 2008 年 9 月 14 日收到修改稿)

提出一种模糊边界模块化神经网络(FBMNN)的混沌时间序列预测方法,该方法先对混沌时间序列观测点重构的相空间进行模块化划分,划分点的选取由遗传算法自动寻优,然后定义一个模糊隶属度函数,在划分边界一侧按照一定的模糊隶属度设定模糊边界带,通过模糊化处理,解决了各模块划分点附近预测结果的跳跃问题,最后每一模块,及其模糊边界的样本点都对应一个递归神经网络进行训练,通过预测合成模块输出结果,该方法对三个混沌时间序列基准数据集 Mackey-Glass, Lorenz, Henon 进行实验,结果表明该方法有效地提高了混沌时间序列预测效果.

关键词:模糊边界,模块化神经网络,混沌时间序列,预测

PACC: 0545

1. 引 言

随着非线性混沌动力学理论和应用研究的不断深入,混沌时间序列的预测成为一个非常重要的研究方向.目前的研究多数是基于 Takens^[1]的嵌入定理和相空间重构思想,其中基于实际观测数据的相空间重构,由标量观测数据构造延迟坐标,得到以相空间点映射的系统动态,并利用神经网络的非线性映射能力,通过学习输入输出样本的基础上获得观测系统模型的方法能对低维混沌时间序列进行较精确地预测.

近年来,模糊神经网络^[2,3],径向基函数神经网络^[4,5],支持向量机^[6-11],回声状态网络^[12],递归型神经网络^[13-15]以及动态时延网络^[16,17]等都被应用于混沌时间序列预测与建模研究中,取得了较好的应用效果.但是,单一神经网络方法往往存在两个问题:第一,单个神经网络在处理问题规模较小,输入节点及隐层节点较少,网络结构不是很复杂时较为适用,而在处理复杂非线性问题时,收敛速度慢,耗时长,泛化能力差;第二,运用 Takens 重构理论对混

沌时间序列的观测数据进行重构后的相空间一般都是高维的,对混沌时间序列预测而言,要通过输入—输出这些高维的相空间点对神经网络进行训练,调节神经网络去模拟混沌系统行为的特定轨迹,显然如果用单个神经网络去模拟这种复杂的非线性系统,随着网络结构复杂程度增加,隐层数和隐层节点数便会增加,其计算复杂性将会很高,并且由于混沌系统本身的特性,往往对单步预测比较准确,而当预测步长逐渐增大,预测精度会急剧降低.

多神经网络是近年来引起国内外研究者浓厚兴趣的一种方法,其中可以分为两类,一类是神经网络集成(neural network ensemble)^[18],它是用有限个神经网络对同一问题进行学习,集成在某输入示例下的输出由构成神经网络集成的各神经网络在此示例下的输出共同决定,适合问题规模不大,每个网络都能对该问题单独处理的情况. Assaad 等^[19]提出将递归神经网络集成的方法进行时间序列单步和多步的预测,取得了比 BPTT(backpropagation trough time)方法更好的效果.另一类是模块化神经网络(modular neural networks)^[20],它是将复杂的问题分解成若干个子问题,然后用相应的网络来处理对应的问题,最

*国家自然科学基金重点项目(批准号:30230350),广东省自然科学基金(批准号:07006474),广东省科技攻关项目(批准号:2007B010200044)资助的课题.

[†] E-mail: qianlima@scut.edu.cn

后再将各个子网络的结果进行合成,这样就大大简化了计算,适合规模较大,计算复杂,容易采用“分而治之”的原则处理的问题.对于混沌时间序列预测问题来讲,模块化神经网络能够将问题求解空间分层,相比单个神经网络提高了泛化能力^[21].Ortín 等^[22]曾提出过用二模块神经网络来对非线性动力学系统进行同步及预测,得到了比前馈网络更好结果;Melin 等^[23]提出用模块化神经网络进行消费品价格方面的预测并取得了较好的预测结果;Goh 等^[24]提出级联模块化复值递归神经网络模型(CPRNN)对 Mackey-Glass 及风速进行了预测,得到了较好的效果.但是,如何将一个混沌时间序列的预测问题分解为若干个合理的、复杂程度降低的子问题,特别是如何自动进行模块化划分而不是通过先验知识来划分,实现问题的自动分解,以及如何选取适当的子网络来对子问题进行学习并且最后通过合成模块输出结果等问题一直是应用模块化神经网络对混沌系统研究的关键.

本文提出一种模糊边界模块化神经网络(FBMNN)的混沌时间序列预测方法.该方法通过对混沌时间序列观测点重构的相空间样本点进行自动模块化划分,划分点的选取由遗传算法自动寻优,并且定义了一个模糊隶属度函数,在划分边界一侧按照模糊隶属度大小设定模糊边界宽度,通过模糊化处理,解决了划分点附近预测结果的跳跃问题.然后每一部分划分出的样本点及其模糊边界带的样本点都对应一个递归神经网络模块进行训练,最后各模块的预测结果通过合成模块组合输出.实验表明,该方法对 Mackey-Glass, Lorenz, Henon 混沌时间序列进行单步和多步预测,不仅能够减少单个网络隐层节点个数,降低计算复杂度,而且预测精度也比参考文献中各种单神经网络及多神经网络方法有所提高,表明了该方法的有效性.

2. 模糊边界模块化神经网络

2.1. 模块化方法

模块化方法是采用“分而治之”的思想,将复杂的问题分解为简单的若干子问题.将其思想引入到神经网络领域,就是将单个神经网络不能或者难以处理的一个复杂问题分解为有限个子问题,使每个子问题能够通过一个比较简单的网络模块来进行处

理,最后将各模块处理结果进行整合,从而使复杂问题得到解决.如图 1 所示.

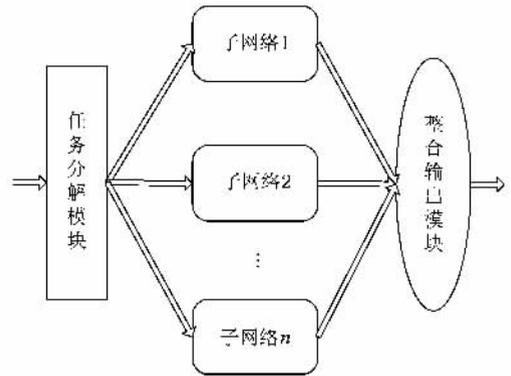


图 1 模块化神经网络结构示意图

参照 Krogh 和 Vedelsby^[25]的神经网络集成的泛化误差计算公式,给出模块化神经网络的泛化误差公式.

设整个模块化神经网络结构一共由 n 个子网络共同组成,对 $f:R^m \rightarrow R$ 进行近似,子网络的权重为 $w_i, i=1, 2, \dots, n$, 并且 $\sum_{i=1}^n w_i = 1$, 单个子网络的输出为 $V_i(X), i=1, 2, \dots, n$, 整个网络的输出为

$$\bar{V}(X) = \sum_{i=1}^n w_i V_i(X). \quad (1)$$

子网络 i 的泛化误差 E_i 为

$$E_i = (V_i(X) - f(X))^2. \quad (2)$$

各网络的泛化误差加权平均为

$$\bar{E} = \sum_{i=1}^n w_i E_i. \quad (3)$$

子网络 i 的差异度 A_i 为

$$A_i = (V_i(X) - \bar{V}(X))^2. \quad (4)$$

整个模块化神经网络的差异度加权平均为

$$\bar{A} = \sum_{i=1}^n w_i A_i. \quad (5)$$

整个模块化神经网络的泛化误差 E 为

$$E = (\bar{V}(X) - f(X))^2, \quad (6)$$

则有

$$E = \bar{E} - \bar{A}. \quad (7)$$

由(7)式可知,要使整个模块化神经网络的泛化误差较小,就要尽可能使模块间的差异度增大,使每个子网络的误差互不相关.

因此,模块化神经网络方法一般涉及到三个方面的问题:

1) 如何将一个复杂问题分解为有限个合理的子问题,使得子问题能比原问题处理简单;

2) 如何选择子模块对应每个子问题,并且针对不同的子问题复杂程度和规模,选择不同的子网络模块,使模块间差异度增大;

3) 各模块间如何通信,包括训练时模块间的配合以及最后对每个子网络模块预测结果的合成.

针对任务分解问题, Lu 等^[26]在研究模块化神经网络应用到模式分类问题时,将模块化分解方法分为三类:直接分类法、类分解法、自动分解法.前两种方法都需要设计者对某领域的专业知识,而第三种方法则不需要先验知识,在学习的过程中自动进行模块化划分.这种方法最方便,也是研究人员致力于研究的目标.

本文的模块化分解方法采用自动分解法.首先将训练数据分为两部分:一部分用于每个子网络训练,另一部分用于整个模型的验证.然后对划分点进行编码,利用遗传算法寻找最优划分点,即初始划分点将重构后的相空间点分成若干子空间,如果重构后的相空间有 m 维,则初始划分点将相空间分为 2^m 个子空间,每个子空间中的点对应一个子网络模块去训练.此时的训练只是部分训练 (partial training),训练回合数为 500.当所有子网络模块都训练结束后,得出该划分点在验证数据上的误差,对误差排序后,淘汰误差较大的后 30% 的划分点,然后再补充新的随机生成的划分点,进入下一代,直到得到预测效果最优的划分点.设有时间序列 $x_t, t = 1, \dots, N$. 则计算模块化划分点的算法描述如下.

第一步 根据时间序列观测点重构相空间,选择适当的嵌入维 m 和延迟时间 τ ,得到相空间点 X_t 为

$$X_t = (x_{t-(m-1)\tau}, x_{t-(m-2)\tau}, \dots, x_t) \in R^m. \quad (8)$$

第二步 设模块化分界点为

$$D = (x_{t-(m-1)\tau}, x_{t-(m-2)\tau}, \dots, x_t) \in R^m. \quad (9)$$

将 D 随机生成多个初始值,产生不同的个体,形成遗传算法的初始种群,通过遗传算法的变异算子 p_m 和交叉算子 p_c 对个体进行交叉变异操作,划分点 D 就把相空间分为 2^m 个子空间.

第三步 将各个子空间中的点输入到子网络中,进行部分训练,训练结束后计算每个子模块 i 在验证数据 (validation data) 上的误差 e_i ,将 $1/e_i$ 作为评价网络个体好坏的适应度函数 $fitness_i$.

第四步 根据适应度函数值排序,利用遗传算法选择算子 p_s 选择适应度较高的部分个体,返回第二

步,连同再次随机生成的划分点个体,形成新的种群,进入下一代进化,直到达到最大进化代数或者适应度满足条件,算法结束.

上述算法第二步中,若重构相空间的嵌入维数为 m ,则划分点 D 会把相空间分为 2^m 个子空间,当 $m \geq 6$ 时,子空间 (或子网络) 个数会达到 64 个以上,但通过对多个混沌时间序列实验发现,被划分出来的很多子空间中是不包含任何数据点的,因此,实际参加计算的子网络个数一般都是小于 20 的,再加上此时的训练只是部分训练,所以当嵌入维数 $m \geq 6$ 时,训练效率并不会有明显降低.

2.2. 边界模糊化

按照上节所述得到合适的划分点,再通过训练数据对每个子网络进行训练结束后,输入预测数据得到结果会发现预测结果在划分边界附近的点会出现偏离较大、甚至跳跃的现象,这是由于有些相空间点处在分界线上,这些点数量较少,有可能只有几个,而它们却被相邻两个网络训练,从而导致预测得出的点偏差较大,在相空间中表现为这些点所在的轨迹在穿过模块边界时发生偏离.要解决这个问题,就涉及到模块间如何通信以及最后对每个子网络的预测结果进行合成的问题.

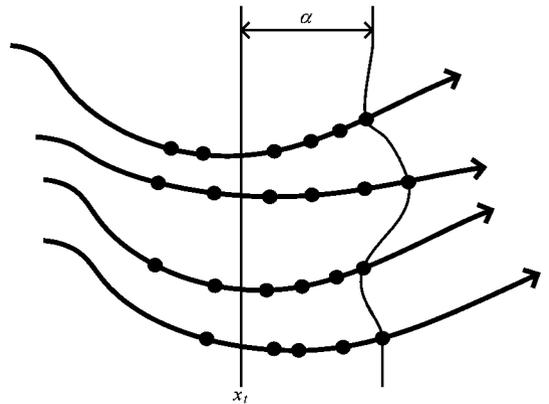


图 2 模糊边界

本文采用模糊化方法,通过定义一个模糊隶属度函数,在划分点一侧按照模糊隶属度大小划定模糊宽度,从而便在原来的划分边界处形成了一个不规则的模糊边界带,将边界带中的点按照其隶属度划分到与其相邻的子空间当中,这样就避免了边界处数据点太少而导致预测结果跳跃问题,平滑了边界,并且使得训练时模块间通过模糊隶属度联系起来,也有利于最后对每个子网络模块预测结果的合

成.模糊边界示意图如图 2 所示.

给出模糊边界定义如下:设分界点为

$$D = (x_{t-(m-1)\tau}, x_{t-(m-2)\tau}, \dots, x_t),$$

对吸引子的某一条轨迹,假定 x_t 为分界点,给定一个正整数 N ,有序列 $\{x_{t+1}, x_{t+2}, \dots, x_{t+N}, \dots\}$,则对相空间某一维度,定义模糊边界宽度为

$$\alpha = \max\{|x_{t+1} - x_t|, |x_{t+2} - x_t|, \dots, |x_{t+N} - x_t|\}, \quad (10)$$

即 $\{x_{t+1}, x_{t+2}, \dots, x_{t+N}\}$ 是沿着 x_t 所在的轨迹方向且距离 x_t 最近的 N 个点,对 $\forall x_{t+p}$,当 $P > N$ 时,有

$$|x_{t+p} - x_t| > \alpha,$$

当 $P \leq N$ 时,有

$$|x_{t+p} - x_t| \leq \alpha,$$

则对穿过某一子空间的所有轨迹来说,每条轨迹的 α 都不一定等长,因此边界是不规则的,如图 2 所示.我们将从 x_t 开始,以 α 为宽度的不规则区域叫做模糊边界带.对某一模块 M_i 中的某一维数据点 x ,定义其模糊隶属度函数如下:

$$A(x) = \begin{cases} 1, & x \leq x_t + \alpha, \\ 0, & x > x_t + \alpha. \end{cases} \quad (11)$$

同理可以扩展到高维.

这样,就在分界点 x_t 一侧形成了包含 N 个数据点、宽度为 α 的不规则模糊边界带.在训练阶段,对于相邻两个子模块 M_i 和 M_{i+1} ,模糊边界中的点 x 既会划分到 M_i 中进行训练,也会划分到 M_{i+1} 中进行训练.而在预测阶段,每个数据点有且仅有一个子模块对其进行计算.最后,整合输出模块通过判断每个数据点的隶属度将每个子模块输出的数据整合起来,得到最后的结果.

2.3. FBMNN 结构

本文提出的 FBMNN,其结构包括相空间重构、自动分解模块、子网络模块以及整合输出模块四大部分.首先选择适当的嵌入维 m 和延迟时间 τ 得到重构相空间,然后经过自动分解模块将相空间划分为若干子空间,并在划分边界一侧按照一定的模糊宽度设定模糊边界带,每一部分划分出的样本点及其模糊边界带的样本点都对应一个递归神经网络模块进行训练.最后各模块的预测结果通过整合输出模块组合输出,其整体结构如图 3 所示.

图 3 中,递归子网的灰色区域便是本文定义的模糊边界带,相邻子空间都会有部分数据处于模糊

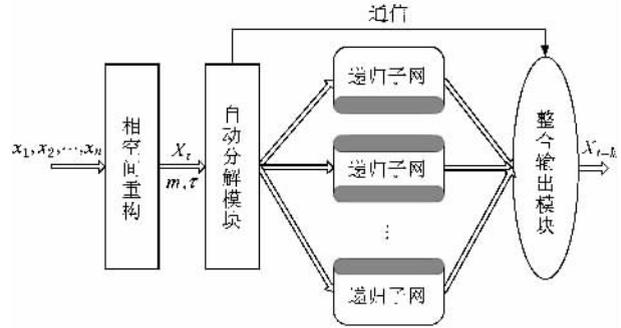


图 3 FBMNN 结构

边界带中.

在模块化方法中,要注意模块间的通信和协作.本文方法就是通过自动分解模块与整合输出模块之间进行通信,来确保训练数据在被分到不同的各个子网络并训练完毕后,测试数据能够根据其子在子空间中的隶属度,被正确地分配到相应的子网络中去计算,最后能“整合还原”输出预测序列结果.

针对如何选择子网络的问题,由于递归神经网络(recurrent neural networks)是动态网络,能够通过内部的递归连接反映非线性动力系统的特性,在时间序列预测中的用途很大^[27],因此本文采用的子网络是 Elman 递归网络^[28],它是一种典型的内反馈型递归神经网络. Elman 网络一般是三层组成:输入层、隐含层、输出层,其中隐层节点的输出会反馈到输入层的一些反馈节点(context units).反馈节点用来记忆隐层前一时刻输出值,可以把它看作是一个延迟算子.前馈部分由连接权修正,递归部分固定不变.因此,Elman 网络具有递归性和动态性.如图 4 所示.

设输入为 u ,隐层输出为 x ,反馈节点输出为 x_c .网络输出为 y ,则网络的非线性状态空间表达式如下:

$$\begin{aligned} x(k) &= f(W_1 x_c(k) + W_2 u(k-1)), \\ x_c(k) &= x_c(k-1), \\ y(k) &= g(W_3 x(k)), \end{aligned} \quad (12)$$

式中 W_1, W_2, W_3 分别为反馈节点到隐层节点、输入节点到隐层节点、隐层节点到输出节点、隐层节点到反馈节点的连接权.隐层节点到反馈节点的连接权值固定为 1. $f(\cdot)$ 和 $g(\cdot)$ 分别为输出层和隐层的传递函数. $f(\cdot)$ 通常采用 S 型函数,而 $g(\cdot)$ 采用线性函数.

由 (7) 式可知,要使整个模块化神经网络的泛化误差减小,就要尽可能使模块间的差异度增大.为了

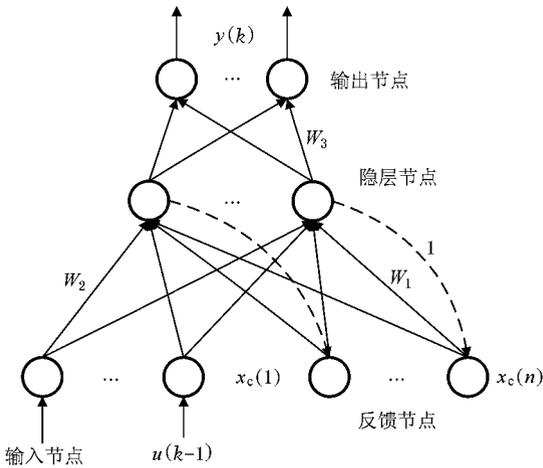


图4 Elman 网络结构

增大模块间的差异度,我们采用非均衡子网策略,即分界点将相空间分解后,针对不同的子空间中数据点的多少采用不同结构的子网络,对数据点较多的子空间,对应子网络隐藏节点较多,结构相对复杂;对数据点较少的子空间,对应子网络隐藏节点较少,结构相对简单,即所谓“大空间”对应“大网络”;“小空间”对应“小网络”。通过非均衡子网策略,增大了模块间的差异度,提高了模块化方法的灵活性,并增强了对不同混沌系统处理的适应性。

根据上述分析,本文提出的基于模糊边界模块化神经网络方法,其算法的整个流程描述如下。

1)相空间重构:由混沌时间序列的观测点数据选择适合的嵌入维 m 和延迟时间 τ ,并根据它们生成模型所需的输入向量 $X(N \times m$ 维)和输出向量 $Y(N \times 1$ 维)。

2)将整个数据集分为三部分:训练数据、验证数据、测试数据,三部分数据量的比例为 14:6:5。

3)采用部分训练计算整个模型在验证数据集上的适应度值,利用遗传算法选择分界点,自动将重构后的相空间划分为有限个子空间,其中遗传算法变异算子 p_m 取 0.1,交叉算子 p_c 取 0.8,种群大小取 20,最大进化代数取 50。

4)分界点确定后,对各模块间划分边界进行模糊化处理,按照模糊隶属度决定每个边界数据点的归属。

5)将每部分子空间以及模糊边界中的数据输入到对应的 Elman 子网络进行训练,训练采用 Levenberg-Marquardt 算法,训练最大回合数取 2000,并采用非均衡子网策略,当某个子空间中的数据点

个数 $\text{num}_i < 200$ 时,采用的网络隐藏节点个数为 4,当 $200 \leq \text{num}_i < 400$ 时,采用的网络隐藏节点个数为 6,当 $\text{num}_i \geq 400$ 时,采用的网络隐藏节点个数为 8。

6)利用第 5 步中训练好的 FBMNN,根据测试数据所属不同子空间分配到不同子网络,计算测试数据 $X(t)$ 对应的预测值 $X(t+k)$ (k 为预测步长,当 $k=1$ 时为单步预测,当 $k>1$ 时为多步预测),最后通过合成模块输出单步或多步预测结果。

3. 实验结果及其对比分析

为了验证 FBMNN 的有效性,我们选取了 Mackey-Glass, Lorenz, Henon 三个基准数据集进行单步和多步预测,并且和文献中的结果进行了比较。

试验中,对每个基准数据集产生 2500 个数据,并将整个数据集分为三部分:训练数据、验证数据、测试数据,三部分数据量的比例为 14:6:5,即训练数据 1400 个,验证数据 600 个,预测数据 500 个。

为了方便与文献中预测结果进行比较,我们采用均方误差(MSE)、均方根误差(RMSE)、标准化均方误差(NMSE)以及绝对误差 $e(t)$,作为误差评判标准:

$$\text{MSE} = \frac{\sum_{t=1}^S [\hat{y}(t) - y(t)]^2}{S}, \quad (13)$$

$$\text{RMSE} = \sqrt{\text{MSE}}, \quad (14)$$

$$\text{NMSE} = \frac{\sum_{t=1}^S [\hat{y}(t) - y(t)]^2}{\sum_{t=1}^S [y(t) - \bar{y}]^2}, \quad (15)$$

$$e(t) = \hat{y}(t) - y(t). \quad (16)$$

(13)–(16)式中 $y(t)$, $\hat{y}(t)$, \bar{y} 和 S 分别是真实值、预测值、真实值的均值和样本大小。

3.1. Mackey-Glass 的混沌时间序列预测

1977 年发现的 Mackey-Glass 时滞系统经常被用作检验非线性系统模型性能的标准,其方程为

$$\frac{dx}{dt} = a \frac{x(t-\tau)}{1+x^c(t-\tau)} - bx(t), \quad (17)$$

$$\tau = 17, a = 0.2, b = 0.1, c = 10.$$

采用四阶龙格-库塔法寻找方程(17)的数值解,时间步长为 0.1,通过对数值积分,得到时间序列 $x(t)$ 。在重构相空间时,取嵌入维数 $m=4$,延迟时间 $\tau=6$ 。即用嵌入矢量 $X_t = (x_t, x_{t-6}, x_{t-12}, x_{t-18})$ 预测

x_{i+k} , 其中 k 为预测步长. 通过遗传算法寻优得到的划分点是 $D=(0.6 \ 0.7 \ 0.6 \ 0.5)$. 在所划分的 $2^m = 2^4 = 16$ 个子空间中, 只有 6 个子空间有数据, 因此整个模型有 6 个子模块. 取模糊边界宽度 α 为 20, 最后得到 1 步预测结果的 RMSE 与广义 RBF^[4], IC-E^[5], IOC-GB^[5], MFT-SVM^[6], 在线小波 SVR^[9], 以及 MNN^[22] 结果比较如表 1 所示.

表 1 Mackey-Glass 时间序列 1 步预测

方法	RMSE
广义 RBF	6.41×10^{-3}
IC-E	1.07×10^{-2}
IOC-GB	6.5×10^{-3}
MFT-SVM	1.3×10^{-2}
在线小波 SVR	1.2×10^{-3}
MNN2 2	9.04×10^{-4}
MNN4 2	9.41×10^{-4}
FBMNN	3.02×10^{-4}

从表 1 可以看出, 在对 Mackey-Glass 基准数据集进行单步预测中, FBMNN 取得了比其他方法更高的预测精度. 预测效果如图 5 所示. 根据(16)式计算的预测绝对误差如图 6 所示.

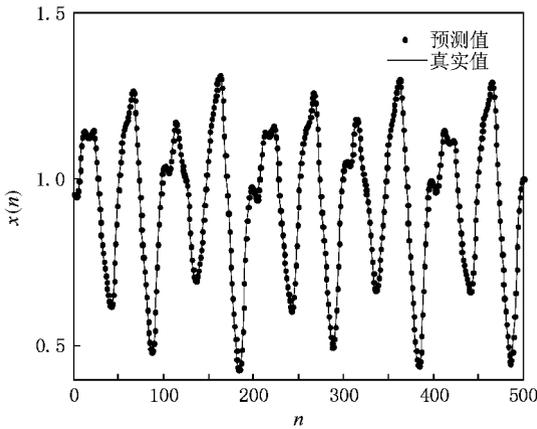


图 5 Mackey-Glass 预测结果($k=1$)

从图 6 的绝对误差可以看出, FBMNN 比文献 [29] 中的绝对误差 $\epsilon(n)$ 小一个数量级.

同时, 我们也做了多步预测的实验, 计算对应的 RMSE 和 NMSE. 其中 6 步预测的 RMSE 与 Local DCT domain Vloterra 预测方法^[30], MFLNN^[15] 和 LLNF^[31] 比较结果如表 2 所示(文献 [30] 中的方法只给出了 MSE, 而表中列出的是经过(14)式计算得到的 RMSE. 下文出现文献 [32] 中的 RMSE 同理).

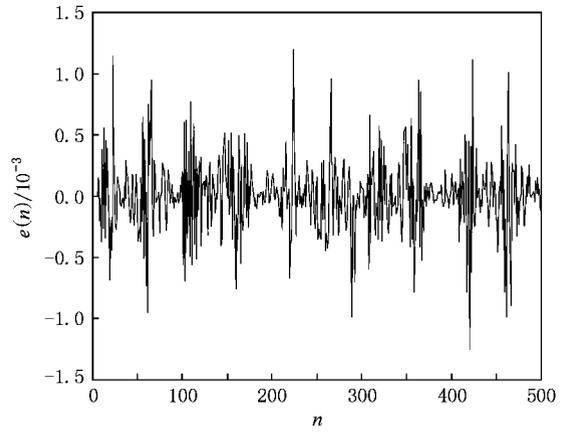


图 6 Mackey-Glass 预测误差($k=1$)

表 2 Mackey-Glass 时间序列 6 步预测

方法	RMSE
Local DCT domain Vloterra	4.50×10^{-2}
MFLNN	1.5×10^{-3}
LLNF	9.61×10^{-4}
FBMNN	8.79×10^{-4}

在 1—14 步预测中, 与 Boosting 方法^[19] 的 linear150, squared100 和 exponential100 三种情况的 NMSE 比较如表 3 所示, 图 7 画出了 FBMNN 各预测步长对应的 NMSE 曲线.

表 3 FBMNN 与 Boosting 方法的 NMSE 对比

步长	Lin.	Squ.	Exp.	FBMNN
1	1.7×10^{-4}	1.6×10^{-4}	1.7×10^{-4}	1.776×10^{-6}
2	2.4×10^{-4}	2.8×10^{-4}	2.5×10^{-4}	7.42×10^{-6}
3	5.7×10^{-4}	5.7×10^{-4}	5.2×10^{-4}	2.64×10^{-5}
4	5.7×10^{-4}	5.4×10^{-4}	5.2×10^{-4}	5.13×10^{-5}
5	9.8×10^{-4}	1.26×10^{-3}	1.27×10^{-3}	5.20×10^{-5}
6	2.11×10^{-3}	1.525×10^{-2}	4.66×10^{-3}	1.50×10^{-5}
12	6.72×10^{-3}	8.66×10^{-3}	7.57×10^{-3}	5.54×10^{-5}
14	1.521×10^{-2}	2.132×10^{-2}	1.56×10^{-2}	3.11×10^{-5}

从表 3 可以看出, FBMNN 的 14 步以内的预测中比采用 Boosting 算法的递归神经网络集成方法提高了 1—3 个数量级, 同时从图 7 可以看出, 与 Boosting 方法的结果类似, 随着预测步数的增大, NMSE 逐渐增加, 但中间有所反复.

在 10—30 步预测中, FBMNN 与 CPRNN^[24] 模型比较结果如表 4 所示.

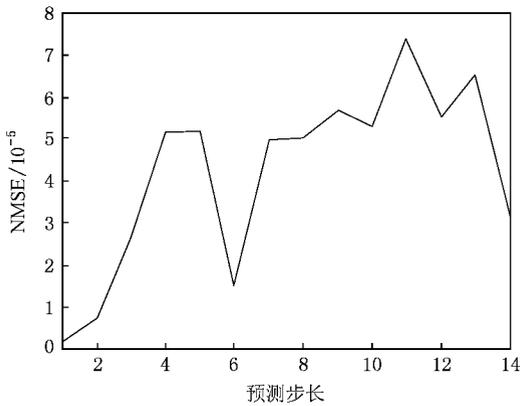


图 7 Mackey-Glass 预测步长与 NMSE 的关系

表 4 FBMNN 与 CPRNN 的 NMSE 对比

步长	CPRNN	FBMNN
10	1.154×10^{-1}	5.25×10^{-5}
20	1.876×10^{-1}	4.54×10^{-3}
30	1.907×10^{-1}	1.33×10^{-2}

同样是采用递归神经网络, CPRNN 模型采用的是级联模块化方式, 而 FBMNN 是采用并联模块化方式. 从表 4 的结果分析, FBMNN 在多步预测的精度上要比 CPRNN 高, 但是随着步长的增大, 精度会逐步降低, 而 CPRNN 预测性能表现得较为稳定, 但是精度不高.

3.2. Lorenz 的混沌时间序列预测

Lorenz 模型的方程为

$$\begin{aligned} \dot{x}_1 &= \sigma(y - x), \\ \dot{y}_1 &= -xz + rx - y, \\ \dot{z}_1 &= xy - bz, \\ \sigma &= 10, r = 28, b = 8/3. \end{aligned} \quad (18)$$

同样采用四阶龙格-库塔法计算求解获得时间序列, 并将 x 分量归一化到 $[0, 1]$ 区间, 取嵌入维数 $m = 6$, 延迟时间 $\tau = 1$. 通过遗传算法寻优得到的划分点是 $D = (0.7, 1, 0.7, 0.9, 0.8)$. 在所划分的 $2^m = 2^6 = 64$ 个子空间中, 只有 7 个子空间有数据, 因此整个模型有 7 个子模块. 取模糊边界宽度 α 为 20, 最后得到 1 步预测结果的 RMSE 与 IC-E, IOC-GB 及最小二乘支持向量域^[10]比较如表 5 所示.

从表 5 可以看出, 在对 Lorenz 基准数据集进行单步预测中, FBMNN 比其他方法的预测精度提高了 2—3 个数数量级. 单步预测效果如图 8 所示, 根据

(16) 式计算的预测绝对误差如图 9 所示.

表 5 Lorenz 时间序列 1 步预测

方法	RMSE
IC-E	1.23×10^{-2}
IOC-GB	8.5×10^{-3}
最小二乘支持向量域	2.2×10^{-3}
FBMNN	2.046×10^{-5}

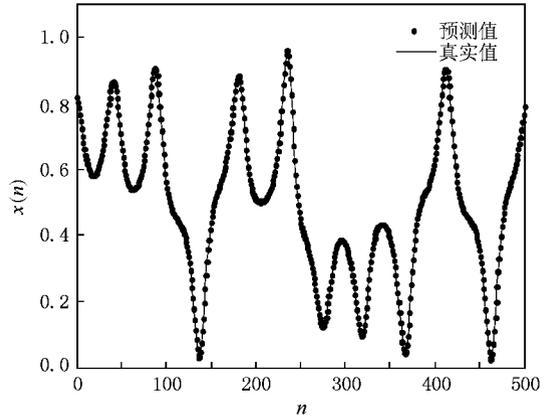


图 8 Lorenz 预测结果($k = 1$)

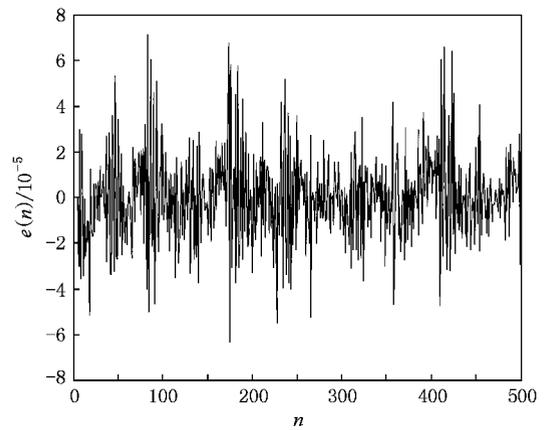


图 9 Lorenz 预测误差($k = 1$)

从图 9 可以看出, FBMNN 的绝对误差 $e(n)$ 的数量级为 10^{-5} , 比文献 [29] 中的 10^{-2} 提高了 3 个数数量级.

同时, 我们也做了多步预测的实验, 预测步长 k 分别取 6 和 8, 其中 6 步预测的 RMSE 与 Local DCT domain Vloterra 预测方法和 SONVK (CG)^[32] 比较结果如表 6 所示.

从表 6 可以看出, FBMNN 方法在 6 步预测中的 RMSE 比其他方法有较大提高. 并且, 在 8 步预测

中 ,FBMNN 的 RMSE 为 0.00048 ,比文献[10]中的 0.0031 的精度更高.

表 6 Lorenz 时间序列 6 步预测

方法	RMSE
SONVK CG)	2.14×10^{-2}
Local DCT domain Vloterra	2.04×10^{-3}
FBMNN	3.7×10^{-4}

3.3. Henon 的混沌时间序列预测

Henon 映射是著名的简单动力学系统之一 ,其方程为

$$\begin{aligned} x(t+1) &= 1 - ax(t)^2 + y(t), \\ y(t+1) &= bx(t), \\ a &= 1.4, b = 0.3, \end{aligned} \quad (19)$$

计算 x 分量数值 ,取嵌入维数 $m = 4$,延迟时间 $\tau = 1$.通过遗传算法寻优得到的划分点是 $D = (0.7, 0.8, 0.5, 0.4)$.在所划分的 $2^m = 2^4 = 16$ 个子空间中 ,有 9 个子空间有数据 ,因此整个模型有 9 个子模块 .取模糊边界宽度 α 为 20 ,最后得到 1 步预测结果的 RMSE 与文献中相关结果比较如表 7 所示 .

表 7 Henon 时间序列 1 步预测

方法	RMSE
广义 RBF	4.13×10^{-2}
在线 LS-SVM 回归 ^[7]	4.4×10^{-3}
最小二乘支持向量域	3.0×10^{-3}
Local DCT domain Vloterra	1.1×10^{-4}
FBMNN	2.7×10^{-5}

图 10 给出了 Henon 实际序列和基于 FBMNN 的预测序列混沌吸引子图 .从吸引子的比较上可以看出 ,本文提出的 FBMNN 对 Henon 混沌时间序列预测是有效的 .

在多步预测中 ,我们取步长 k 为 2,3,4 ,其中 1—3 步预测所得到的结果与文献[31]中的 RBF (OLS) 和 LoLiMoT 比较如表 8 所示 ,其中 FBMNN 中所列为单个模块中隐层节点个数的取值范围 .

从表 8 可以看出 ,1 步预测时 ,FBMNN 比 RBF (OLS) 精度高 ,虽然比 LoLiMoT 精度低 ,但由于每个模块的网络隐层节点数量较少 ,因此计算复杂度有所降低 .而在 2,3 步预测中 ,FBMNN 的精度均比其他两种方法高 .

另外 ,FBMNN 对 Henon 4 步预测的 RMSE 为

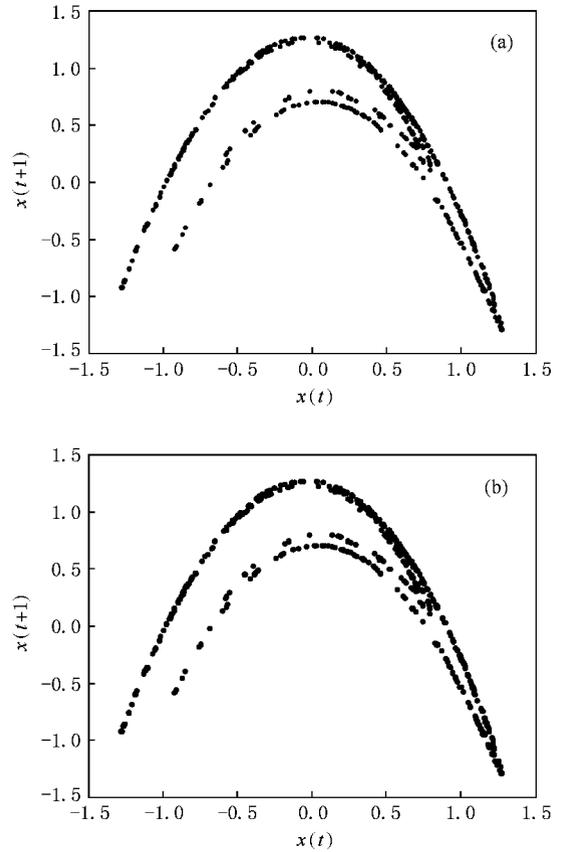


图 10 Henon 时间序列混沌吸引子 (a)实际序列 (b)基于 FBMNN 预测序列

0.0074 ,比 fuzzy sigmoid^[8]的 0.0092 低 .因此在多步预测上 FBMNN 预测效果更好 .

表 8 Henon 预测结果 NMSE 比较

方法	1 步预测	2 步预测	3 步预测
RBF (OLS)	1.521×10^{-9} 35 节点	6.049×10^{-4} 38 节点	8.72×10^{-2} 42 节点
LoLiMoT	3.600×10^{-10} 30 节点	2.065×10^{-6} 30 节点	2.91×10^{-2} 20 节点
FBMNN	1.469×10^{-9} 4—8 节点	3.695×10^{-8} 4—8 节点	1.1×10^{-3} 4—8 节点

4. 结 论

本文提出了一种模糊边界模块化神经网络 (FBMNN) 的混沌时间序列预测方法 .该方法通过对混沌时间序列观测点重构的相空间样本点进行模块化划分 ,划分点的选取由遗传算法自动寻优 ,实现了对混沌时间序列模块化的自动划分 .而通过定义一

个模糊隶属度函数,在划分边界处按照一定的模糊宽度设定模糊边界带,通过模糊化处理,解决了划分点附近预测结果的跳跃问题.同时,本文所提方法能将混沌时间序列预测这一复杂问题有效的划分为若干个子问题,解决每个子问题的网络隐层平均只需

要4—8个节点,有效的降低了计算复杂度,提高了预测精度.对几种混沌时间序列基准数据集的实验可以看出,该方法能对产生混沌时间序列的动力学方程进行逼近,从而有效的进行单步和多步预测.

- [1] Takens F 1981 *Dynamical Systems and Turbulence* (Berlin :Spring-Verlag) 366
- [2] Tan W ,Wang Y N ,Zhou S W ,Liu Z R 2003 *Acta Phys. Sin.* **52** 795 (in Chinese) [谭 文、王耀南、周少武、刘祖润 2003 物理学报 **52** 795]
- [3] Hu Y X ,Gao J F 2005 *Acta Phys. Sin.* **54** 5034 (in Chinese) [胡玉霞、高金峰 2005 物理学报 **54** 5034]
- [4] Li J ,Liu J H 2005 *Acta Phys. Sin.* **54** 4569 (in Chinese) [李 军、刘君华 2005 物理学报 **54** 4569]
- [5] Zhang J F ,Hu S S 2007 *Acta Phys. Sin.* **56** 713 (in Chinese) [张军峰、胡寿松 2007 物理学报 **56** 713]
- [6] Cui W Z ,Zhu C C ,Bao W X ,Liu J H 2005 *Chin. Phys.* **14** 922
- [7] Ye M Y ,Wang X D ,Zhang H R 2005 *Acta Phys. Sin.* **54** 2568 (in Chinese) [叶美盈、汪晓东、张浩然 2005 物理学报 **54** 2568]
- [8] Liu H ,Liu D ,Deng L F 2006 *Chin. Phys.* **15** 1196
- [9] Yu Z H ,Cai Y L 2006 *Acta Phys. Sin.* **55** 1659 (in Chinese) [于振华、蔡远利 2006 物理学报 **55** 1659]
- [10] Ren R ,Xu J ,Zhu S H 2006 *Acta Phys. Sin.* **55** 555 (in Chinese) [任 韧、徐 进、朱世华 2006 物理学报 **55** 555]
- [11] Zhang J S ,Dang J L ,Li H C 2007 *Acta Phys. Sin.* **56** 67 (in Chinese) [张家树、党建亮、李恒超 2007 物理学报 **56** 67]
- [12] Jaeger H ,Haas H 2004 *Science* **304** 78
- [13] Han M ,Xi J H ,Xu S G ,Yin F L 2004 *IEEE Trans. Signal Processing* **52** 3409
- [14] Tan W ,Wang Y N 2005 *Chin. Phys.* **14** 72
- [15] Savran A 2007 *IEEE Trans. Neural Networks* **18** 373
- [16] Xie J X ,Cheng C T ,Yu B ,Zhang Q R 2005 *Artificial Neural Networks :Formal Models and Their Applications* **3697** 565
- [17] Becerikli Y ,Oysal Y 2007 *Applied Soft Computing* **7** 1164
- [18] Sollich P ,Krogh A 1996 *Advances in Neural Information Processing Systems* **8** 190
- [19] Assaad M ,Boné R ,Cardot H 2008 *Information Fusion* **9** 41
- [20] Happel B L M ,Murre J M J 1994 *Neural Networks* **7** 985
- [21] Allende H ,Salas R ,Torres R ,Moraga C 2005 *Advances in Soft Computing* **2** 585
- [22] Ortín S ,Gutiérrez J M ,Pesquera L ,Vasquez H 2005 *Phys. A* **351** 133
- [23] Melin P ,Mancilla A ,Lopez M ,Mendoza O 2007 *Applied Soft Computing* **7** 1217
- [24] Goh S L ,Chen M ,Popovic D H ,Aihara K ,Obrodovic D ,Mandic D P 2006 *Renewable Energy* **31** 1733
- [25] Krogh A ,Vedelsby J 1995 *Advances in Neural Information Processing Systems* **7** 231
- [26] Lu B L ,Ito M 1999 *IEEE Trans. Neural Networks* **10** 1244
- [27] Duda R O ,Hart P E ,Stork D G 2005 *Pattern Classification ,Second Edition* (Beijing :China Machine Press) p265 (in Chinese) [李宏东、姚天翔译 2005 模式分类(机械工业出版社)第265页]
- [28] Elman J L 1990 *Cognitive Science* **14** 179
- [29] He T ,Zhou Z O 2007 *Acta Phys. Sin.* **56** 693 (in Chinese) [贺 涛、周正欧 2007 物理学报 **56** 693]
- [30] Zhang J S ,Li H C ,Xiao X C 2005 *Chin. Phys.* **14** 49
- [31] Gholipour A ,Araabi B N ,Lucas C 2006 *Neural Processing Letters* **24** 217
- [32] Li H C ,Zhang J S ,Xiao X C 2005 *Chin. Phys.* **14** 2181

Chaotic time series prediction based on fuzzy boundary modular neural networks^{*}

Ma Qian-Li[†] Zheng Qi-Lun Peng Hong Qin Jiang-Wei

(College of Computer Science and Engineering ,South China University of Technology ,Guangzhou 510640 ,China)

(Received 17 October 2007 ; revised manuscript received 14 September 2008)

Abstract

A fuzzy boundary modular neural network (FBMNN) is proposed for the chaotic time series prediction. First ,the reconstructed phase space is divided into several subspaces and the divided points are evaluated by genetic algorithms. Then a fuzzy membership function is defined and the fuzzy boundary is set on the border according to the fuzzy membership. Through this fuzzy treatment ,the jumping problem of the predicted data near the divided points are solved. Finally the data points of each module and its fuzzy boundary are input to a recurrent neural network for training and the output predicted points are synthesized by a synthesis forecast module. The effectiveness of FBMNN is evaluated by using three benchmark chaotic time series data sets : the Mackey-Glass series ,Lorenz series ,and Henon series. The simulation results show that FBMNN improves the performance of chaotic time series prediction.

Keywords : fuzzy boundary , modular neural network , chaotic time series , prediction

PACC : 0545

^{*} Project supported by the State Key Program of National Natural Science Foundation of China (Grant No. 30230350) ,the Natural Science Foundation of Guangdong Province ,China (Grant No. 07006474) ,the High-tech Program of Guangdong ,China (Grant No.2007B010200044).

[†] E-mail :qianlima@scut.edu.cn