

含噪混沌时间序列预测误差分析*

王新迎 韩敏[†] 王亚楠

(大连理工大学, 电子信息与电气工程学部、控制科学与工程学院, 大连 116023)

(2012年6月11日收到; 2012年10月26日收到修改稿)

对于含噪混沌时间序列预测问题, 传统方法存在较大的经验性, 对预测误差的构成分析不足, 因而忽略了混沌动态重建与预测模型建立之间的差异性. 本文将实际预测误差分解为预测器偏差和输入扰动误差, 并对整体最小二乘和正则化两种全局预测方法进行分析比较, 进而说明整体最小二乘适用于混沌动态的重建, 对预测器偏差影响较大, 而正则化方法能够改善预测器敏感性, 对输入扰动误差影响较大. 通过两个仿真实例, 展示了混沌动态重建与预测模型建立之间的差异, 在对比最小二乘和正则化方法的同时验证了实际预测误差受预测器偏差和输入扰动误差共同作用. 并指出, 在实际操作时应二者间寻求平衡, 以便使模型预测精度达到最优.

关键词: 混沌时间序列预测, 噪声, 整体最小二乘, 正则化

PACS: 05.45.Tp

DOI: 10.7498/aps.62.050504

1 引言

通过分析时间序列的历史记录建立预测模型, 不仅可以分析研究时间序列的内在机理, 而且有助于相关事件的决策支持^[1-3]. 近年来, 涉及混沌时间序列预测的研究较多, 提出了多种混沌时间序列预测方法: 如利用 RBF 网络^[4,5], 多项式拟合^[6], 递归神经网络^[7-9], 支持向量机^[10-12] 等对混沌时间序列建模预测. 但是大多数预测方法的研究对象主要为不含噪声的混沌时间序列. 由于实际观测的混沌时间序列通常包含噪声干扰, 因而导致随机性增加, 制约着预测的精度. 因混沌系统遵循自身的运行规律, 可通过非线性噪声消减方法削弱时间序列的随机性, 增强其确定性^[13-16], 进而改善整体预测效果^[5]. 但在某些情况下, 由于记录长度和初始噪声水平的限制, 噪声消减的作用有限, 处理后的时间序列仍可能包含一定量的噪声.

噪声的存在将会对混沌时间序列的预测效果产生不利影响^[17]. 而现有预测模型通常是在常规最小二乘方法基础上建立的, 抗噪性不强; 同时, 在对含噪混沌时间序列进行预测时欠缺对预测误差

的全面分析, 不利于预测精度的进一步提高. 一方面, 由于系统输入输出数据中均包含噪声^[18], 噪声环境下的混沌动力学重建是一个变量含误差模型问题, 因此对混沌动态的恢复是一个棘手的问题. 另一方面, 当将预测器应用于实际问题时, 预测模型的输入不可避免地受到噪声的影响, 由于混沌系统内在的初始状态敏感性, 在某些条件下由输入噪声通过预测器产生的误差往往是不可忽视的, 同时由于混沌系统特有的非线性, 这部分误差在相空间中的分布是不均匀的, 而目前的研究较多地忽略了这部分的误差.

目前, 有些研究者采用整体最小二乘的方法对含噪混沌序列进行无偏重建^[19,20], 以获取更为准确的混沌演化规律. 通常的观点也认为只要得到精确的演化规律就能得到好的预测结果. 然而, 该方法仅着重考虑了第一个方面引起的误差, 而忽视了第二个方面的误差, 限制了预测精度的进一步提高. 针对这个问题, 本文从实际预测误差上界出发, 给出预测器偏差和输入扰动误差的定义, 同时通过考察不同建模方法对实际预测结果的影响, 分析给出实际预测误差的构成及其决定因素.

* 国家自然科学基金 (批准号: 61074096) 资助的课题.

[†] 通讯作者. E-mail: minhan@dlut.edu.cn

2 含噪混沌时间序列的预测问题

假设一不含噪声的混沌时间序列为 $\{x(t), t = 1, 2, 3, \dots\}$, 通过延迟坐标重构得到未含噪的相点 $\mathbf{d}(t)$ 为

$$\mathbf{d}(t) = [x(t), x(t - \tau), \dots, x(t - (m - 1)\tau)]^T,$$

其中, m 和 τ 分别为混沌时间序列的嵌入维数和延迟时间.

考虑混沌时间序列的 h 步预测关系 $\mathbf{d}(t) \rightarrow x(t+h)$, 希望找到预测器 $\mathbf{f}(\cdot)$, 使得

$$x(t+h) = \mathbf{f}(\mathbf{d}(t)),$$

其中预测器 $\mathbf{f}(\cdot)$ 可以通过神经网络逼近得到, 网络方程可以表示为

$$x(t+h) = \mathbf{f}_x(\mathbf{w}, \mathbf{d}(t)).$$

这里, \mathbf{f}_x 为某一个神经网络预测器 (下文中出现 $\mathbf{f}_A, \mathbf{f}_B, \mathbf{f}_C$ 三种), \mathbf{w} 为神经网络的可调连接权值.

当混沌时间序列本身包含噪声时, 会使得重构相空间中的相点偏离其真实位置, 预测器构建过程和实际预测过程都会受到噪声的影响, 使得实际预测过程的误差变得难以分析.

若假设时间序列在 t 时刻受到的噪声影响记为 $\xi(t)$, 相应的实际观测值为 $s(t) = x(t) + \xi(t)$, $t = 1, 2, 3, \dots$, 经相空间重构受扰后的相点 $\mathbf{d}(t)$ 变成相点 $\mathbf{s}(t)$ 为

$$\mathbf{s}(t) = [s(t), s(t - \tau), \dots, s(t - (m - 1)\tau)]^T.$$

含噪相点 $\mathbf{s}(t)$ 的偏移量可表示为

$$\begin{aligned} \boldsymbol{\varepsilon}(t) &= \mathbf{s}(t) - \mathbf{d}(t) \\ &= [\xi(t), \xi(t - \tau), \dots, \xi(t - (m - 1)\tau)]^T. \end{aligned}$$

假设序列 $\{\boldsymbol{\varepsilon}(t), t = 1, 2, 3, \dots\}$ 为独立同分布、均值为零的随机噪声, 并设定含噪混沌时间序列的噪声水平为 v (定义为 $v = \text{std}(\xi)/\text{std}(x)$), 那么由 $E[x^2] = \text{var}[x]$, 得到

$$E[\|\boldsymbol{\varepsilon}\|^2] = m \cdot \text{var}[\xi] = m \cdot v^2 \cdot \text{std}^2(x),$$

即偏移量 $\boldsymbol{\varepsilon}$ 的数学期望

$$E(\|\boldsymbol{\varepsilon}\|) = \sqrt{m} \cdot \text{std}(x) \cdot v.$$

特别地, 当随机噪声为高斯白噪声时, 随机变量 $\|\boldsymbol{\varepsilon}\|^2/(v^2 \cdot \text{std}^2(x))$ 服从自由度为 m 的卡方分布.

同时, 对于含噪混沌时间序列, 构建预测器所用的训练样本实际上变为

$$\{\mathbf{s}(t) \rightarrow s(t+h), t = 1, 2, 3, \dots\}.$$

而不再反映未染噪声情况下 $\mathbf{d}(t)$ 与 $x(t+h)$ 之间的对应关系. 此外, 神经网络预测器的实际输出变为 $\mathbf{f}_x(\mathbf{w}, \mathbf{s}(t))$, 而不再是 $\mathbf{f}_x(\mathbf{w}, \mathbf{d}(t))$.

此时, 含噪混沌序列的实际预测误差 E_p 可以表示为

$$\begin{aligned} E_p &= \sum_t \|x(t+h) - \mathbf{f}_x(\mathbf{w}, \mathbf{s}(t))\| \\ &= \sum_t \|\mathbf{f}(\mathbf{d}(t)) - \mathbf{f}_x(\mathbf{w}, \mathbf{s}(t))\|, \end{aligned} \quad (1)$$

其中 \mathbf{f} 反映了理想的混沌动态. 可以看出 \mathbf{f} 是定义在未含噪的相点之上, 而预测器 \mathbf{f}_x 不得不对受扰预测相点 $\mathbf{s}(t)$ 进行响应.

对预测误差 E_p 进行范数不等式扩展得

$$\begin{aligned} E_p &= \sum_t \|\mathbf{f}(\mathbf{d}(t)) - \mathbf{f}_x(\mathbf{w}, \mathbf{d}(t)) + \mathbf{f}_x(\mathbf{w}, \mathbf{d}(t)) \\ &\quad - \mathbf{f}_x(\mathbf{w}, \mathbf{s}(t))\| \\ &\leq \sum_t \|\mathbf{f}(\mathbf{d}(t)) - \mathbf{f}_x(\mathbf{w}, \mathbf{d}(t))\| + \sum_t \|\mathbf{f}_x(\mathbf{w}, \mathbf{d}(t)) \\ &\quad - \mathbf{f}_x(\mathbf{w}, \mathbf{s}(t))\| \\ &= E_b + E_n. \end{aligned} \quad (2)$$

不等式右侧第一项是预测器在未染噪相点上的预测误差, 反映了预测器本身的偏差, 本文定义为预测器偏差 E_b ; 第二项反映了预测器 \mathbf{f}_x 由于输入扰动引起的误差, 定义为预测器的输入扰动误差 E_n . 从 (2) 式可以看出, 当预测输入包含噪声时, 实际预测误差 E_p 的上界与预测器偏差 E_b 和输入扰动偏差 E_n 之和有关.

当混沌时间序列不含噪声时, 有

$$\begin{aligned} \mathbf{s}(t) &= \mathbf{d}(t), \\ \mathbf{f}_x(\mathbf{w}, \mathbf{s}(t)) &= \mathbf{f}_x(\mathbf{w}, \mathbf{d}(t)), \\ E_p &= \sum_t \|\mathbf{f}(\mathbf{d}(t)) - \mathbf{f}_x(\mathbf{w}, \mathbf{s}(t))\| \\ &= \sum_t \|\mathbf{f}(\mathbf{d}(t)) - \mathbf{f}_x(\mathbf{w}, \mathbf{d}(t))\| \\ &= E_b. \end{aligned}$$

此时, 实际预测误差 E_p 等于预测器偏差 E_b , 预测的目标是令预测器最大程度逼近重构相空间. 当混沌

时间序列包含噪声时, 实际预测误差 E_p 的上界与 E_b 和 E_n 都相关, 即使重建所得的预测器 $f_x = f$, 也无法保证使得 E_p 充分小. 又因为预测器 f_x 模拟的是混沌系统本身, 输入扰动误差 E_n 的大小本质上体现了混沌系统初始状态敏感性如何作用于实际预测误差. 下文将分别就预测器偏差 E_b 和输入扰动误差 E_n 进行具体讨论.

3 含噪混沌时间序列预测误差分析

本小节将结合整体最小二乘方法和正则化方法对预测器偏差 E_b 和输入扰动误差 E_n 进行较为具体地分析, 并指出针对实际混沌时间序列预测问题, 需要在两者之间进行折中处理, 进而使预测精度达到最好.

3.1 预测器偏差

为了尽可能地减小预测器偏差 E_b , 希望能够从含噪的输入输出数据中还原出预测关系 f , 使得预测器 f_x 尽可能逼近 f . 由于模型输入也包含噪声, 常规的基于最小二乘的非线性回归方法将得到有偏的预测器估计. 为了对混沌动态实施无偏重建, 非线性回归可以在整体最小二乘的框架下进行.

文献 [15—17] 通过整体最小二乘法来改造神经网络的训练目标函数, 采用目标函数 J_A 进行神经网络训练

$$J_A = \sum_{t=1}^N \left\| \left\| \begin{bmatrix} s(t) \\ s(t+h) \end{bmatrix} - \begin{bmatrix} y(t) \\ f_A(w, y(t)) \end{bmatrix} \right\|^2 \right\|. \quad (3)$$

其中 $y(t)$ 为 t 时刻神经网络的输入, 可以表示为

$$y(t) = [y(t), y(t-\tau), \dots, y(t-(m-1)\tau)]^T,$$

t 时刻的输入变量序列由时间序列变量 $\{y(1), y(2), \dots, y(t), \dots\}$ 得到, 在网络的训练过程中, 神经网络的权值 w 和网络输入 $y(t)$ 都是需要调整的参数, 可以通过梯度下降算法训练得到. 理想情况下每一个矢量 $y(t)$ 将会逼近 $d(t)$, 预测器 f_A 也会很好地逼近真实的混沌动态 f .

从理论上讲, 由目标函数 J_A 得到的预测器可以较为准确地反映混沌系统的动态特性, 有助于进一步研究混沌系统的性质.

考虑另一个方面的问题, 既然预测器 f_A 不得不从含噪序列中提取, 在实际预测过程中, 预测输

入不可避免再会受到噪声的影响, 也就说由 (3) 式得到的预测器偏差仅仅是实际预测误差的组成部分之一.

3.2 预测器输入扰动误差

采用类似于 (3) 式的目标函数可以控制预测器偏差 E_b , 但无法抑制输入扰动误差 E_n . 由于实际含噪混沌序列预测误差与两者均相关, 因此要提高预测精度, 需要兼顾地对 E_b 和 E_n 进行控制.

假设 f_x 为某个混沌系统预测器, 那么对预测原点 $s(t)$ 的扰动产生的输入扰动误差可由泰勒展开表示为

$$E_n = \sum_t \|f_x(w, d(t)) - f_x(w, s(t))\| = \sum_t \|f'_x(w, d(t))\varepsilon(t) + \frac{1}{2!} f''_x(w, d(t))\varepsilon^2(t) + \dots\|, \quad (4)$$

其中 $f'_x(w, d(t))$ 和 $f''_x(w, d(t))$ 分别表示预测器输出对输入的一阶和二阶导数.

由 (4) 式可知, 为了尽可能减小预测输入扰动误差 E_n , 预测器 f_x 对输入不应该有较大灵敏度, 也就是说预测器的输出应该对输入具有一定程度的光滑性. 如果灵敏度过高, 输入微小的扰动就会带来较大的 E_n . 在非线性回归技术中, 通过引入目标函数的高阶导数项可以有效地控制待求预测函数的光滑程度, 这种方法被称之为正则化方法. 考虑如下的目标函数:

$$J_B = \sum_{t=1}^N \|s(t+h) - f_B(w, s(t))\|^2. \quad (5)$$

在目标函数 J_B 中, 预测器的输入为含噪后的相点 $s(t)$, 已有的研究表明在训练神经网络时, 在输入中加入某种幅值的噪声等价于加入 Tikhonov 正则项 [21,22]. 也就是说, 如果采用目标函数 J_B 训练神经网络, 相当于隐式地使用了 Tikhonov 正则化, 因此可以从某种程度上改善预测器响应的光滑性和灵敏度.

由于待处理时间序列所包含的噪声是固定的, 因此使用目标函数 J_B 训练神经网络时, 注入的噪声项无法控制, 起到的正则项作用有限. 针对这个问题, 可以通过引入显式的正则项的方式, 实现对预测器的灵敏度进行控制. 令目标函数 J_C 如下:

$$J_C = \lambda \sum_{t=1}^N \|f_C(w, s(t)) - s(t+h)\|^2$$

$$+ (1 - \lambda) \|Df_C\|^2, \quad (6)$$

其中, D 表示微分操作, $0 \leq \lambda \leq 1$ 为正则项系数, 用于控制正则项的强弱. 可以将目标函数 J_B 看作是 J_C 的一种特殊形式 ($\lambda = 1$). 实际存在的预测方法, 已经存在很多目标函数形如 J_C 的预测模型, 例如目标函数具有权衰减项的多层感知机、正则化 RBF^[2]、核岭回归和支持向量机等.

对于局部线性模型来说, 由于预测函数在局部可以用线性函数表示, 在局部数据集上, 输入扰动误差不存在全局逼近方法中由于非线性函数引起的不平衡性, 因此输入扰动误差 E_n 的讨论仅限于全局模型.

3.3 E_b 和 E_n 对实际预测误差的影响

根据以上分析, 当时间序列不含噪声时, 即 $d(t) = s(t), t = 1, 2, 3, \dots$ 时, 只有当预测器 f_x 准确地重建混沌系统的动力学特性, 即满足 $f = f_x$ 时, 可以令实际预测误差 E_p 为零. 考虑当噪声无限大时的情况, 此时实际输入 $s(t)$ 为无界. 由于混沌吸引子是有界的, 由 f 反映的真实混沌动态在有界的输入 ($d(t)$) 下输出也是有界的, 而 $f = f_x$ 时在输入 $s(t)$ 作用下, 预测器的输出却不能保证在有限的范围内. 也就是说在实际问题中, 小的预测器偏差 E_b 和小的输入扰动误差 E_n 往往是不可兼得的, 必须根据噪声的具体情况, 在两者间寻求适当的平衡点, 才能改善实际预测结果.

在实际建立神经网络预测器的过程中, E_b 和 E_n 的平衡可能发生于:

- 1) 在网络训练过程中, 选择合适的停止时间时;
- 2) 选择合适的正则化系数时.

针对这个问题, 一个有效的方法是交叉检验^[9,12]. 在神经网络的训练过程, 在独立的交叉检验样本计算预测误差, 当该预测误差开始上升时停止训练过程, 这个过程也叫提前停止. 另外在交叉检验样本上计算实际预测性能, 并以此为依据选择恰当的正则化系数.

需要指出的是, 交叉检验样本也来源于含噪混沌时间序列, 在此样本上得到的预测误差指标衡量的既不是预测器偏差 E_b , 也不是输入扰动误差 E_n , 而是实际预测误差 E_p . 换句话说, 找到的“最优”预测器是在 E_p 上界最小的意义下的, 而非为预测器偏差 E_b 最小.

此外, 采用交叉检验方法还可以考察混沌系统的无偏重建和实际预测的差异性, 即通过预测器偏

差 E_b 和输入扰动误差 E_n 的对比关系, 研究在目标函数 J_A 和 J_B 得到预测器的差异性. 其中, J_A 是以得到较小的 E_b 为目的, 而 J_B 的使用是以得到较小的 E_p 为目的, 目标函数 J_C 作为 J_B 的改进形式, 能够更好地控制正则化的强弱.

4 仿真实例

在以下的仿真实例中, 首先在 Logistic 映射的基础上考察一种极端的情形, 即当预测器偏差 E_b 为零时, 实际预测误差 E_p 与相点位置的关系, 从而说明输入扰动误差 E_n 在相空间中的不平衡性; 其次, 对含噪 Henon 时间序列建立多种预测模型, 考察在不同的建模过程中, 预测器偏差 E_b 和输入扰动误差 E_n 在实际预测误差 E_p 中所占的比例, 从而研究实际预测误差的组成和决定因素.

4.1 E_b 和 E_n 对实际预测误差的影响

Logistic 映射如下式所示:

$$x(k+1) = ax(k)(1-x(k)). \quad (7)$$

当 $a = 4$ 时, 利用该方程可产生混沌时间序列. Logistic 映射如图 1 所示.

(7) 式的右端实际是个 Logistic 映射的准确单步预测器, 即

$$x(t+1) = f(x(t)) = 4x(t)(1-x(t)). \quad (8)$$

因为 f 为准确的预测器, 当预测器输入不含噪声时, 得到的预测误差为零. 现在考察当输入包含噪声的情况: 令预测器输入 $s(t) = x(t) + \xi(t), t = 1, 2, 3, \dots, N$. 仿真中把 $x(t)$ 的区间 $[0, 1]$ 平均分为 20 子区间, 在每个子区间的端点位置加入 5% 高斯白噪声 $\xi(t)$, 并通过公式 $|x(t+1) - 4s(t)(1-s(t))|$ 计算输入扰动误差 E_n (此时, E_n 等于实际预测误差 E_p). 这样的过程随机进行 2000 次, 将得到的输入扰动误差绘制误差棒图如图 2 所示. 从图中可以看出, 输入扰动误差在不同区域得到的误差水平是不同的, 当 x 位于中间 0.5 左右时, 输入扰动误差 E_n 的均值和方差较小; 当 x 位于两端 0, 1 附近时, 输

入扰动误差 E_n 具有较大的均值和方差.

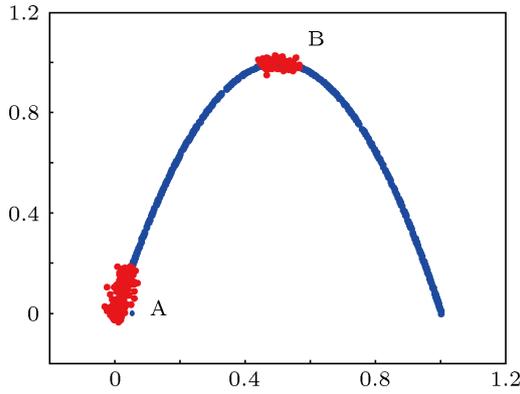


图1 Logistic映射上染噪的区域A和区域B

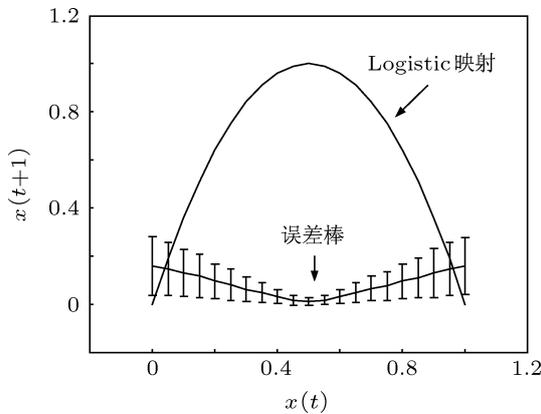


图2 使用 Logistic 映射预测染噪序列的误差棒图

输入扰动误差 E_n 在相空间内的不平衡性可以用预测器的导函数进行解释, 容易算出无偏预测器 f 的各阶导数分别为

$$\begin{aligned} df &= 4(1-2x)dx, & d^2f &= -8dx^2, \\ d^n f &= 0, & n &\geq 3. \end{aligned}$$

那么由 x 的扰动引起的预测误差经泰勒展开可以近似地表示为

$$\begin{aligned} \Delta e &\approx \left| df + \frac{1}{2!}d^2f + \frac{1}{3!}d^3f + \dots \right. \\ &= 4(1-2x)dx - 4dx^2|. \end{aligned} \quad (9)$$

由 (9) 式容易看出, 当 f 相对于 x 的灵敏度越大时, 由预测器产生的预测误差 E_p (E_n) 就越大. 在图 1 中, 由于区域 A 的灵敏度高于区域 B, 因此在区域 A 处预测风险高于区域 B. 从这个简单的例子可以看出, 即便获得了准确的预测器, 由于输入扰动的的影响, 实际预测误差仍可能非常大. 在实际预测过程中, 输入扰动误差 E_n 很可能占实际预测误差 E_p 的主要部分.

4.2 含噪 Henon 时间序列的仿真结果

本小节将分别采用整体最小二乘和正则化方法建立神经网络预测器, 并在模型建立过程中研究预测器的不同性质和决定实际预测误差的相关因素. 训练数据采用长度为 2000 的染高斯白噪声 Henon 时间序列, 噪声水平为 20%. 使用染噪的时间序列进行相空间重构, 选择嵌入维数 $m = 4$, 延迟时间 $\tau = 1$, 多层感知机的输入变量个数为 4, 对应 4 个延迟坐标分量. 另外独立产生长度为 500 的 Henon 时间序列, 得到相应的未染噪和染噪的测试相点 (其中染噪的测试集中的噪声仍为 20% 高斯白噪声). 仿真中的所有误差均是在这 500 个独立于训练样本的时间序列上计算所得.

4.2.1 整体最小二乘方法和隐式的正则化方法

仿真中使用多层感知机逼近预测器 f_x , 首先研究目标函数 J_A 和 J_B 的差异性, 即整体最小二乘和隐式正则化方法应用于含噪混沌时间序列预测的差异性. 采用目标函数 J_A 和 J_B 训练多层感知机网络, 保持两种方法的网络结构相同 (单个隐层, 隐层神经元个数为 8, 输出神经元个数为 1), 并通过随机数种子控制两种方法神经网络的初始权值相同. 采用梯度下降算法训练网络的可调权值, 并保持网络学习步长相同 (采用固定步长).

计算并绘制出在目标函数 J_A 和 J_B 作用下的预测器偏差 E_b 和输入扰动误差曲线 E_n , 如图 3 所示 (对数坐标). 在图 3 中曲线 1 和 2 分别对应两种预测器的输入扰动误差 E_n , 曲线 3 和 4 对应为预测器偏差 E_b . 其中曲线 1 和曲线 4 分别是在目标函数 J_A 作用下的 E_n 和 E_b , 可以看出在目标函数 J_A 的整体最小二乘作用下, 预测器偏差 E_b (曲线 4) 远小于在目标函数 J_B 作用下的预测器偏差 (曲线 3), 与此同时, 其输入扰动误差 (曲线 1) 呈上升趋势且远高于在 J_B 作用下的输入扰动误差 (曲线 2), 而曲线 2 则呈现出先上升后下降的趋势. 这是因为, 采用目标函数 J_A 的目的是为了寻找精确的混沌动态, 但对预测器的敏感度没有约束, 因次导致预测器偏差 E_b 不断下降, 而输入扰动误差 E_n 不断上升; 目标函数 J_B 由于输入噪声产生的正则化作用, 使输入扰动误差 E_n 受到抑制 (曲线 2), 与此同时影响到预测

器偏差 E_b (曲线 3).

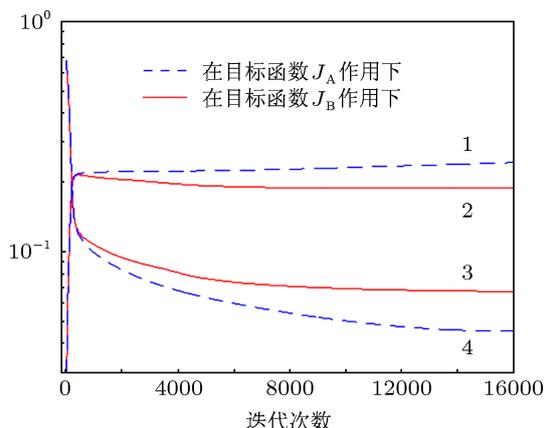


图3 预测器偏差 E_b 和输入扰动误差 E_n 曲线

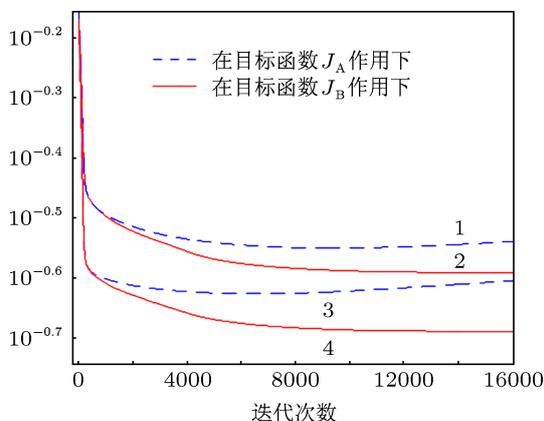


图4 不等式右端项 $(E_b + E_n)$ 与实际预测误差

为了研究两种方法得到实际预测误差 E_p , 在图 4 中绘制 (2) 式不等式右端项 $(E_b + E_n)$ 和实际预测误差 E_p . 曲线 1 和 3 代表在目标函数 J_A 作用下的 $E_b + E_n$ 和 E_p , 曲线 2 和 4 对应由目标函数 J_B 作用下 $E_b + E_n$ 和 E_p 的取值. 从图 4 中可以看出实际预测误差 E_p 与 $E_b + E_n$ 之和具有相似的变化规律, 也就是说其上界从某种程度上体现了实际预测误差的大小.

对照图 3 和图 4 可以可知, 整体最小二乘方法能够得到较为准确的混沌动态, 获得较小的预测器偏差 E_b , 但是由于其输入扰动误差 E_n 数值较大, 使得实际预测误差的上界 $(E_b + E_n)$ 较大, 如图 4 中曲线 1 所示; 而目标函数 J_B 的隐式正则化作用, 使得输入扰动误差得到抑制, 从而使实际预测误差的上界较小, 得到的实际预测误差 E_p 也较小, 分别如图 4 中曲线 2 和曲线 4 所示.

采用交叉检验方法确定合适训练停止时间, 对于整体最小二乘方法来说, 图 4 中实际预测误差曲

线 3 开始上升的时间是实际预测误差 E_p 达到最小的时间, 而并非是预测器无偏重构效果最好的时间. 实际上, 衡量无偏重构好坏的指标预测偏差 E_b 并没有达到最优, 如图 3 中曲线 4 所示.

文献 [13, 14] 的目的是尽可能地减小预测器偏差 E_b (图 3 中曲线 4), 但与之相对应的输入扰动误差 E_n 已经在实际预测误差的组成中占了决定性的比重 (图 3 中曲线 1).

4.2.2 显式正则化方法

由于目标函数 J_B 的隐式正则化作用, 能够对输入扰动误差 E_n 具有较好的抑制作用. 为了能够控制正则化的强弱, 这里显式地使用目标函数 J_C , 并通过调整正则项系数的大小来定量的控制正则化作用. 仿真中 J_C 具体采用常见的权衰减目标函数 J_{CW} , 如下式:

$$J_{CW} = \lambda \sum_{t=1}^N \|f_{CW}(\mathbf{w}, s(t)) - s(t+h)\|^2 + (1-\lambda) \frac{1}{K_w} \|\mathbf{w}\|^2, \quad (10)$$

其中, K_w 为网络连接权值的个数. 本例中研究正则项系数如何影响实际预测误差的构成, 分别选取正则项系数 $\lambda = 0.970, 0.980, 0.990, 0.992, 0.995, 0.997, 0.999$ 和 $\lambda = 1$ (对应于隐式的正则项) 进行仿真. 对每个 λ 的不同取值, 保持神经网络的结构不变, 改变随机数的种子产生 30 组不同的初始权值, 训练每一组神经网络直到网络的权值几乎不发生变化、训练误差不再下降时停止. 注意这里并没有采用交叉检验确定合适停止时间, 其目的是让神经网络完全地受到该正则化水平的作用, 进而能够考察正则化对实际预测误差构成的影响, 与“训练”

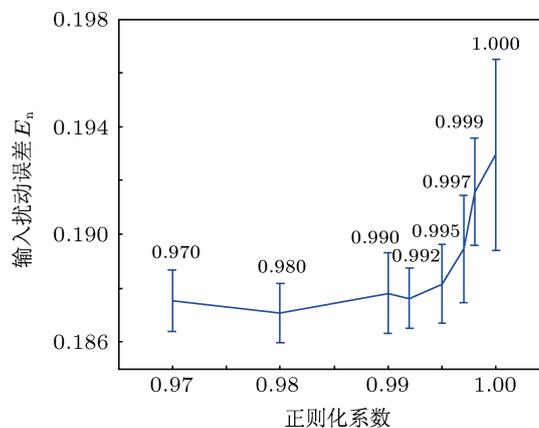


图5 输入扰动误差 E_n 随 λ 的变化情况

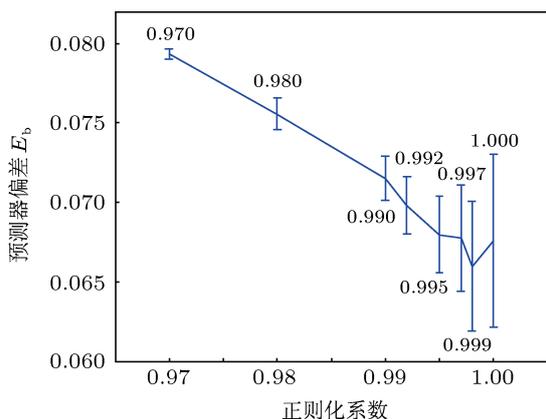


图6 预测器偏差 E_b 随 λ 的变化情况

得到较好的结果 (均值和方差均较小).

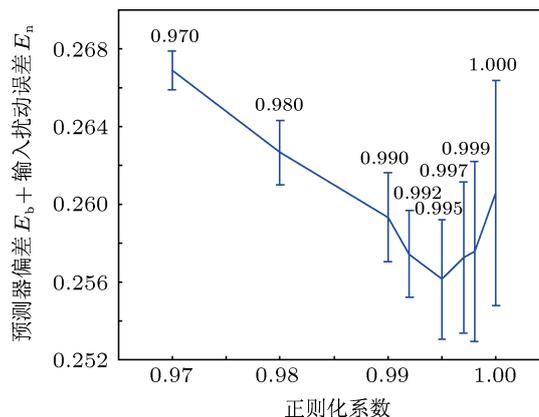


图7 $E_n + E_b$ 随 λ 的变化情况

和“泛化”问题考虑的角度不同.

在测试集合上计算出每个训练结果的输入扰动误差 E_n , 预测器偏差 E_b . 相对于每一个正则化系数 λ 计算误差均值和标准差, 绘制出误差棒图, 如图 5 和图 6 所示. 图 5 为输入扰动误差 E_n 随正则项的变化情况, 图 6 为对应的预测器偏差 E_b 随正则项的变化情况. 从图 5 可以看出, 当正则项系数从 1 逐渐减小时 (正则项水平逐步加强) 输入扰动误差逐步减小, 这说明正则项的作用使得预测器的灵敏性降低, 从而降低输入噪声对输入扰动误差 E_n 的影响; 另一方面, 由于受到正则化的作用, 预测器偏差 E_b 则随着正则项的增强而逐渐增大, 这是正则化对预测器施加的抑制作用. 由此可见训练结果中, 预测器偏差和输入扰动误差相互牵制.

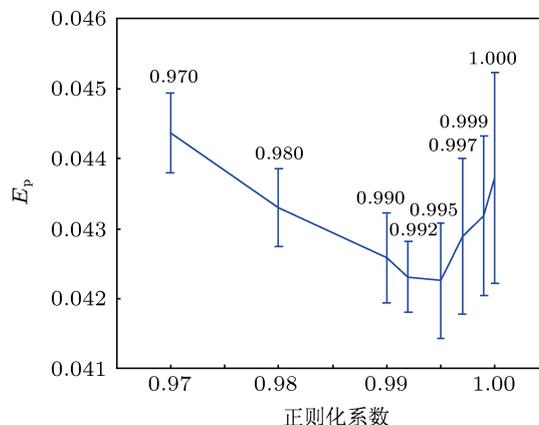


图8 E_p 随 λ 的变化情况

图 7 为预测器偏差 E_b 与输入扰动误差 E_n 之和随正则项系数 λ 变化的误差棒图, 图 8 为实际预测误差 E_p 随正则项系数 λ 变化的误差棒图. 比较图 7 和图 8 可知, 无论是从均值还是从标准差方面考虑, 含噪混沌序列的实际预测误差 E_p 与其上界 $E_b + E_n$ 均具有相似的变化趋势. 这也正说明, 如果想得到较小的实际预测误差 E_p , 需要有效地平衡预测器偏差 E_b 和输入扰动误差 E_n .

5 结论

针对现有的含噪混沌时间序列预测研究容易忽略混沌动态重建与预测模型建立间的差异问题, 本文对实际预测误差的构成进行分析, 并考察了不同的建模方法对实际预测误差的影响, 引入了预测器偏差和输入扰动误差的定义. 将预测器与理想混沌动态之间的误差定义为预测器偏差 E_b , 将由于模型输入含噪声, 受扰的输入引起的误差定义为输入扰动误差 E_n .

当预测器偏差 E_b 较小时, 受混沌系统本身复杂性的影响, 输入扰动误差 E_n 可能主导了实际的预测误差 E_p ; 而当输入扰动误差 E_n 较小时, 由于预测器受到较大的约束, 从而可能使预测器偏差 E_b 在实际预测误差 E_p 中起主导作用. 从图 7 和图 8 中可以看出, 当正则项系数取 0.992 与 0.995 之间的值时, 含噪 Henon 时间序列的实际预测误差可以

针对预测器偏差 E_b 和输入扰动误差 E_n 的不同特性, 分别采用不同方法加以控制. 对仿真结果进行分析, 说明采用整体最小二乘法可以减小 E_b , 获得较为准确的混沌动力学特性. 但由于混沌系统的非线性以及噪声影响的不平衡性, 整体最小二乘法易导致预测器灵敏度过高, 因而实际误差 E_p 仍较大. 而正则化方法可以改善预测器本身的灵敏性, 从而减小 E_n . 当直接用含噪的输入输出数据建

立预测器时, 相当于向预测器施加了隐式的正则化作用; 采用显式的正则化作用可以控制正则化的强度, 从而可以更好地控制输入扰动误差.

由于预测器偏差和输入扰动误差共同影响着实际预测误差, 在实际应用中可以通过交叉检验方法在二者之间进行折中, 以获得最好的实际预测效果.

-
- [1] Karunasinghe D S K, Liang S Y 2006 *J. Hydrol.* **323** 1
- [2] Zhao P, Xing L, Yu J 2009 *Phys. Lett. A* **373** 25
- [3] Molkov Y I, Mukhin D N, Loskutov E M, Timushev R I, Feigin A M 2011 *Phys. Rev. E* **84** 3
- [4] Leung H, Lo T, Wang S 2001 *IEEE Trans. Neural Netw.* **12** 5
- [5] Chen D Y, Liu Y, Ma X Y 2012 *Acta Phys. Sin.* **61** 10 (in Chinese) [陈帝伊, 柳焯, 马孝义 2012 物理学报 **61** 10]
- [6] Su L 2010 *Compu. Math. Appl.* **59** 2
- [7] Jaeger H, Haas H 2004 *Science* **304** 5667
- [8] Song Q S, Feng Z R, Li R H 2009 *Acta Phys. Sin.* **58** 7 (in Chinese) [宋青松, 冯祖仁, 李人厚 2009 物理学报 **58** 7]
- [9] Li D C, Han M, Wang J 2012 *IEEE Trans. Neural Netw. Learn. Syst.* **23** 5
- [10] Zhang J F, Hu S S 2008 *Acta Phys. Sin.* **57** 5 (in Chinese) [张军峰, 胡寿松 2008 物理学报 **57** 5]
- [11] Lau K, Wu Q 2008 *Pattern Recognit.* **41** 5
- [12] Shi Z W, Han M 2007 *IEEE Trans. Neural Netw.* **18** 2
- [13] Gu H, Wang H 2007 *Appl. Math. Comput.* **185** 2
- [14] Gao J B, Sultan H, Hu J, Tung W W 2010 *IEEE Signal Process. Lett.* **17** 3
- [15] Gan J C, Xiao X C 2004 *Chin. Phys.* **13** 3
- [16] Feng J C 2005 *Chin. Phys. Lett.* **22** 8
- [17] Zhang J S, Xiao X C 2000 *Chin. Phys.* **9** 6
- [18] Van G J, Schoukens J, Pintelon R 2000 *IEEE Trans. Neural Netw.* **11** 2
- [19] Lee C C, Chiang Y C, Shih C Y, Tsai C L 2009 *Expert Syst. Appl.* **36** 3
- [20] Lei M, Meng G 2008 *Chaos Solitons Fract.* **36** 2
- [21] Bishop C M 1995 *Neural Comput.* **7** 1
- [22] An G Z 1996 *Neural Comput.* **8** 3

Analysis of noisy chaotic time series prediction error^{*}

Wang Xin-Ying Han Min[†] Wang Ya-Nan

(Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian 116023, China)

(Received 11 June 2012; revised manuscript received 26 October 2012)

Abstract

For the noisy chaotic series prediction problem, traditional methods are quite empirical, and are lacking in the analysis of the composition of the prediction error, thereby ignoring the the difference between chaotic dynamics reconstruction and prediction model. Based on the composition of actual prediction error, the predictor bias error and input disturbance error are defined in this paper and two kinds of global forecasts, ensemble least-square method and regularization method are analysed. It is shown that the ensemble least-square method is suitable for the reconstruction of chaotic dynamics, but has a greater influence on the predictor error. On the other hand, the regularization method can improve the sensitivity of the predictor, but it can be influenced by the input perturbation error. Two simulation examples are used to demonstrate the difference between the chaotic dynamical reconstruction and the establishment of prediction model, and to compare the ensemble least-square method and the regularization method, and at the same time indicate that the actual prediction error is influenced both by the input disturbance error and by the predictor error. In practice, a balance should be stricken between the two, in order to optimize the model prediction accuracy.

Keywords: chaotic time series prediction, noise, ensemble least square, regularization

PACS: 05.45.Tp

DOI: 10.7498/aps.62.050504

^{*} Project supported by the National Natural Science Foundation of China (Grant No. 61074096).

[†] Corresponding author. E-mail: minhan@dlut.edu.cn