基于人工神经网络的赤潮预测模型

马玉梅[1]，高静宇[2]，王清华[3]

（1. 大连民族学院理学院，大连 1166001；2. 河北师范大学计算中心，石家庄 0500162；3. 博涵前锋科技（大连）有限公司 1166002）

摘 要：本文利用非线性时间序列预测模型，将海洋预报和人工神经网络 BP 算法相结合，提出了基于神经网络的海洋预报模型；运用改进的三层 BP（Back Propagation）神经网络模型对海洋气象进行赤潮灾害监测和预报；同时针对仿真结果进行分析，结果表明该模型具有较好的预测能力。

关键词：赤潮；人工神经网络；环境因子；赤潮预报

中图分类号：P731 **文献标识码**：A **文章编号**：1003 － 0239（2007）1 － 0038 － 07

1 引言

赤潮（Red tide）通常是指由于海洋环境条件的变化导致浮游生物（微藻，原生动物或细菌）暴发性增殖或高度聚集使局部水体改变颜色的生态异常现象。由于引发赤潮的生物种类和数量的不同，水体会呈现出不同的颜色，多为红色或砖红色，也称为红潮，也可以是黄色、绿色、棕色或棕红色。海洋浮游微藻是引发赤潮的主要生物，在 4 000 多种浮游微藻中有 260 多种能形成赤潮，中国沿海的赤潮生物有 148 种，其中 43 种曾引发过赤潮[1~2]。

本文利用人工神经网络中的 BP 网络，建立赤潮生物密度与环境因子的人工神经网络的预报模型。以各种理化因子：水温、溶解氧、盐度、总氮、可溶性无机磷、浮游植物密度等为参数，试验人工神经网络的预报效果。结果证明采用人工神经网络进行赤潮预报是行之有效的。

2 人工神经网络的拓扑结构

人工神经网络（Artificial Neural Networks）是由大量的简单神经元广泛连接而成的复杂网络。它是在现代生物学研究人脑组织的基础上提出来的，可用来模拟人类大脑神经的思维活动[4]。它具有并行分布的信息处理结构，通过对非线性函数的复合来逼近输入和输出之间的映射。它不需要设计任何数学模型，只靠过去的经验来学习，通过神经元的模拟、记忆和联想，处理各种模糊的、非线性的、含有噪声的数据，采用自适应的模

收稿日期：2006-01-09；修订日期：2006-12-11

作者简介：马玉梅（1962-），女；教授，主要研究方向：应用数学。

式识别方法来进行预报分析。

神经网络的拓扑结构也是神经网络的一个重要特征。他是由多个神经元通过某种连接方式而组合到一起的，神经网络发展到今天，虽然已经产生了数十种网络模型，但已有的网络大致可分为三类：即前馈网络，反馈网络，还有自组织网络，下图列出了前馈网络和反馈网络的拓扑结构（见图 1）：

BP 算法是前馈网络最重要且应用最普遍的学习算法。由 BP 算法（反向传播算法）训练的多层前馈网络，是神经网络分类器最普遍最通用的形式之一，并且已经证明的基本结论是：由一个隐含层和非线性激励函数组成的网络，能够以任意精度逼近任意的函数。

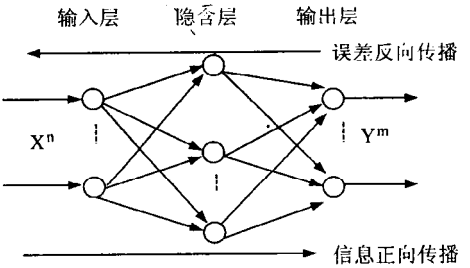

图 1　具有一个隐层的前馈网

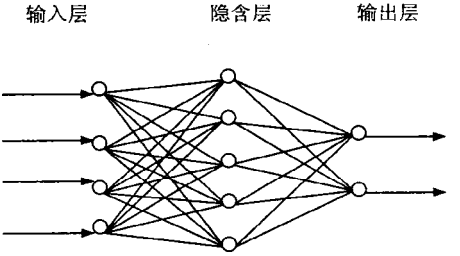

图 2　带有一个隐层的 BP 神经网络模型

3　网络结构设计及网络学习

BP 算法的基本思想是在网络的学习过程中使用梯度搜索技术[3]；利用误差向后传播来修正权值，从而实现网络的实际输出与期望输出的均方差最小化。图 2 给出了通常的带有一个隐含层的 BP 神经网络模型：

作为隐含层和输出层的活化函数，通常采用 Sigmoid 函数，即：

$$f(u)=\frac{1}{1+e^{-u}} \tag{1}$$

对于多层计算和输出按下式计算隐单元 j 的输出为：

$$V_j=f\left(\sum_{k=0}^{K-1}W_{jk}\xi_k-\theta_j\right)\quad j=0,\ 1,\ \cdots,\ J-1 \tag{2}$$

输出单元 i 的输出为：

$$O_i=f\left(\sum_{j=0}^{J-1}W_{ij}V_j-\theta_i\right)\quad i=0,\ 1,\ \cdots,\ I-1 \tag{3}$$

θ_j 表示中间层第 j 个节点的内部阈值，θ_i 表示输出层第 i 个节点的内部阈值。

使用迭代方法一直反向计算到网络的隐含层为止，其计算按由 Delta 规则变形的权值调整方程来进行的。

$$W_{ij}(n+1)=W_{ij}(n)+\eta\,\delta_j Out_j \tag{4}$$

其中$W_{ij}(n)$为神经元 i 至神经元 j 的第 n 次变更的权值；Out_j 为神经元 j 的输出；η 为学习率常数；δ_j 为神经元 j 的差值。

上述第 4 步是一个关键,决定如何向减小差错方向调整网络权值。考察差错的平方和：

$$E=\frac{1}{2}\sum_j(Out_j-\overline{Out_j})^2 \tag{5}$$

为了整个训练集的差错最小，应有全网络平均差错平方和最小。

$$E_{总}=\frac{1}{2p}\sum_p\sum_j(Out_j-\overline{Out_j})^2 \tag{6}$$

其中 p 为训练范例集中训练对的个数。

D. Rumelhart, Hinton 和 Williams1986 年介绍了一个 B-P 算法改进训练时间的方法，即再加一称为“动量”的调整项，调整公式可改为：

$$\Delta w_{ij}(n+1)=\eta\delta_j Out_j+\alpha(w_{ij}(n)-w_{ij}(n-1))=\eta\delta_j Out_j+\alpha[\Delta w_{ij}(n)] \tag{7}$$

$$w_{ij}(n+1)=w_{ij}(n)+\Delta w_{ij}(n+1) \tag{8}$$

这里的 α 是动量常数，0<α<1，通常在 0.9 左右。

当$E_{总}\le\varepsilon$ (ε为预先给订的误差) 时，网络停止训练，此时网络模型就是我们所需的。

4 基于提高精度的动量校正法

因为 BP 算法存在着收敛速度慢，易陷入局部极小等缺点，为使算法具有更大的实用性，必须改进 BP 算法。如何加速 BP 网络的收敛速度和尽量避免最小点是目前研究的热点问题。本文采用一种快速的 BP 算法，它对经典 BP 算法做如下改进，变量说明同上。

每一次对连接权或输出阈值进行校正时，按一定比例加上前一次学习时的校正量，即动量项，由此加速网络学习的收敛速度。具体作法为：

$$w_{ij}(k+1)=w_{ij}(k)+\mu\delta_j(k)x_j(k)+\alpha\Delta w_{ij}(k) \tag{9}$$

式中，$w_{ij}(k+1)$为本次校正量，$w_{ij}(k)$ 为前次校正量，α 为动量因子 (0<α<1)。可知：当前一次的校正量过调时，动量项与本次计算所得误差校正项符号相反，使得本次实际校正量减少，起到减小振荡的作用。而当前一次校正量欠调时，动量项与本次计算所得误差校正项符号相同，本次实际校正量增加，起到加速校正的作用。

5 网络结构设计及数据结构定义

5.1 ANN 预处理及样本的选定

本文通过对夜光藻密度和各环境因子进行分析采用了 28 个样本数据。表 1 选取的理化因子为水温、溶解氧、盐度、总氮、可溶性无机磷、浮游植物密度、夜光藻密度[2]。

表 1 夜光藻密度和各种理化因子的数据[2]

样本	水温（℃）	溶解氧 (mg/L)	盐度	总氮 (mol/L)	可溶性无机磷 (mol/L)	浮游植物密度 (10^4个/m^3)	夜光藻密　度 (10^3个/m^3)
1	23.00	7.14	30.30	0.27	0.43	147.00	7.56
2	23.30	7.22	30.40	0.72	0.10	71.50	530.00
3	23.70	7.36	30.50	2.05	0.36	125.00	260.00
4	23.00	7.06	30.30	0.40	0.54	109.00	2.80
5	23.50	7.38	30.30	0.31	0.47	205.00	4.04
6	23.10	7.12	30.20	0.17	0.40	74.5	5.99
7	23.20	7.31	30.40	0.73	0.26	160.00	466.98
8	24.70	7.56	30.20	0.32	0.32	10.50	91.48
9	22.50	6.97	30.40	0.52	0.56	187.00	7.60
10	23.30	7.54	30.50	0.52	0.07	32.00	180.00
11	23.30	7.44	30.60	0.81	0.18	31.10	440.10
12	23.10	7.50	30.40	0.56	0.39	146.00	469.98
13	22.90	7.06	30.30	0.40	0.54	148.00	1.77
14	23.55	7.56	30.34	0.84	0.08	25.52	169.15
15	22.87	7.12	30.26	0.35	0.47	105.33	3.16
16	24.26	7.82	30.51	0.17	0.41	124.37	7.24
17	23.13	7.40	30.61	0.78	0.19	30.78	414.99
18	23.56	7.38	30.31	0.29	0.54	170.02	3.05
19	22.75	7.61	30.73	0.81	0.24	10.49	24.66
20	22.98	7.19	30.25	0.25	0.37	131.57	7.31
21	23.03	7.12	30.36	0.31	0.41	138.22	7.49
22	22.99	7.44	30.40	0.60	0.34	118.54	309.75
23	22.94	7.17	30.17	0.17	0.41	67.21	9.03
24	23.28	7.57	29.56	0.24	0.48	194.67	92.85
25	24.20	7.83	30.50	0.14	0.40	148.00	8.28
26	22.90	7.56	30.80	0.83	0.21	9.72	23.70
27	23.40	7.54	29.50	0.23	0.43	226.00	60.40
28	23.40	7.62	30.40	0.88	0.07	31.80	159.96

为了满足网络输入输出对数据的要求，在学习之前首先对输入数据按下式进行归一化处理[5]

$$\overline{x_i}=\frac{x_i-\min(x_i)}{\max(x_i)-\min(x_i)} \qquad i=1,\ 2,\ \cdots\cdots,\ m \tag{10}$$

5.2 初始权值与隐含层数的确定

初始权值的确定方法： 初始权值是不应完全相等的一组值。已经有人证明， 即便确定存在一组互不相等的使系统误差更小的权值，如果所设 W_{ji} 的初始值彼此相等，它们将在学习过程中始终保持相等。 故而，在程序中，我们设计了一个随机发生器程序，产生一组-1～1 之间的随机数，作为网络的初始权值。最小训练速率的确定方法：在经典的 BP 算法中，训练速率是由经验确定，训练速率越大，权重变化越大，收敛越快；但训练速率如果过大的话，会引起系统的振荡，因此，训练速率在不导致振荡前提下，越大越好。根据前人经验，一般取该值的大小为 0.5 左右。误差选择：选择输出层每个节点的平方型误差最大为 0.05。

本文用 MATLAB 语言的程序来确定隐含层单元的个数，该网络的输入层的神经元个数为 7，输出层的神经元个数为 1，根据隐含层设计经验公式，以及考虑预测的实际情况，解决该问题的网络的隐含层神经元个数应该在 18～29 之间。因此，下面设计一个隐含层神经元数目可变的 BP 网络，通过误差对比，确定最佳的隐含层神经元个数，并检验活化函数对网络性能的影响。

表 2 网络误差比较

神经元个数	17	18	19	20	21
网络误差	0.1767	0.1449	0.1807	0.1447	0.2642
神经元个数	22	23	24	25	26
网络误差	0.6437	0.1446	0.1442	0.1448	0.1856

表 2 表明，在经过 2000 次训练后，隐含层神经元为 24 的 BP 网络对函数的逼近效果最好，因为它的误差最小，而且网络经过 500 次训练就达到了目标误差。隐含层为 20 和 23 的网络误差也比较小，但它们所需要的训练时间比较长。考虑到网络性能的训练速度，将网络隐含层的神经元数目确定为 24。

从表中可以看出，就是并非隐含层神经元的个数越多，网络的性能就越好，在程序中，误差并没有明显地随着隐含层神经元数目的增加而减小的趋势，当神经元的个数从 20 增加到 21 时，误差反而增大了。在表 2 中神经元个数从 25 增加到 26 也观察到了这个现象。

5.3 活化函数的选取

采用不同的活化函数对网络的性能也有影响，比如收敛速度，下面采用不同的活化函数对网络进行训练，并观察结果。

隐含层和输出层神经元的活化函数都为 logsig 对网络进行训练，该函数的学习算法是梯度下降动量法，而且学习速率是自适应的。当隐含层的神经元数目为 24，网络的逼近误差为 0.1442，网络的训练结果见图 5，从图中我们看到网络经过 500 次训练就达到了目标误差。

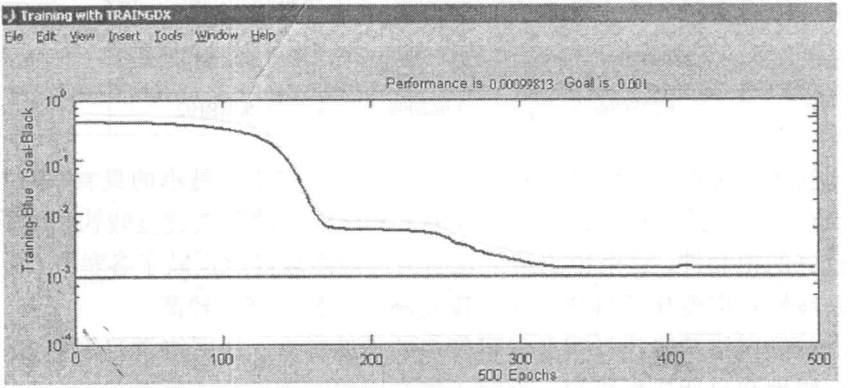

图 5　活化函数为 logsig 的训练结果

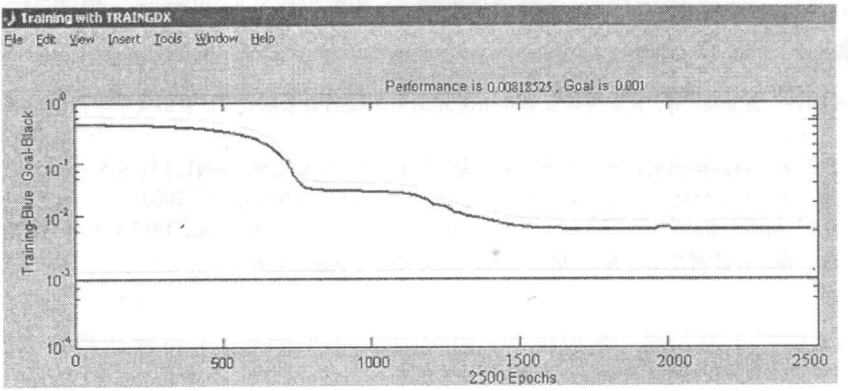

图 6　活化函数为 tansig 的训练结果

最后，网络的隐含层神经元的活化函数采用 tansig，输出层神经元的活化函数采用 logsig，其余代码同图 4。可见，经过 2000 次训练后，网络目标误差依然没有满足，而且通过网络的训练曲线（见图 6），我们看到网络的训练过程收敛的非常缓慢，训练误差比较大 res=0.4146。

至此，我们确定了本预测的最终 BP 网络结构（见表 3）。

表 3　网络结构

网　络　结　构	隐含层神经元	活化函数
单隐层的 BP 网络	24 个	logsig

6　预测结果

使用这个模型，我们选取网络的 24 个样本对网络进行训练之后，输入 4 个样本验

证网络的适应性。所得结果见表 4。

表 4 网络仿真结果误差

	样本 25	样本 26	样本 27	样本 28
实际输出	8.28	23.70	60.40	159.96
网络输出	10.007584	20.637901	66.278102	162.706912
误 差	1.727584	3.062099	5.878102	2.746912

由表 4 的数据可见，网络输出和实际输出的误差较小．最小的只有 1.727584，最大的也只有 5.878102，并且在网络训练过程中，网络误差的收敛速度较快，实际输出和网络出具有较好的拟和性，这个预测结果表明神经网络较好地反映了各种理化因子与夜光藻密度的非线性对应变化的规律，对于夜光藻密度预测精度较高。

一般的神经网络精确模型需要尽可能多的训练样本，让网络学习到更多知识，这会使系统的容错性和可适应性更强。而对于本文网络的训练样本个数仅有 24 个。如果增加训练的样本个数，可以进一步提高网络的非线性映射能力，提高预测精度。

参考文献：

[1] 吴京洪，杨秀环，唐宝英，等．大亚湾澳头增养殖区赤潮与环境的关系研究[J]. 中山大学学报(自然科学版)，2001，40(3)：37～40.
[2] 蔡如钰. 基于人工神经网络的夜光藻密度的预测模型[J]．中国环境监测，2001，17(3)：52～55．
[3] 阎平凡，张长水. 人工神经网络与模拟计划算法[M]. 北京：清华大学出版社，2000.
[4] Lippmann R P. An introduction to computing with neural nets. IEEE ASSP Magazine, 1987,4(2)：4～22.
[5] 李学桥，马 莉．神经网络工程应用[M]. 重庆：重庆大学出版社，1996.

Forecast model for red tide on artificial neural network

MA Yu-mei，GAO Jing-yu, WANG Qing-huan

(1. Natural Science College of Dalian Nationalities University, dalian 116600 China；2. BHR-FRONTLINE Technologies (Dalian) Co. Ltd, dalian 116600 China；3. Compute Center of Hebei Normal University, Shijiazhuang 050016 China)

Abstract：This paper bring forward a forecast model for ocean predict by combining nonlinear time sequence and artificial neural network. Using this model to make a forecast the rid tide of ocean and giving an analysis to the imitate result. It show that the model is provided with better ability for forecast.

Key words：Red tide；Artificial neural network；Environment factors；Prediction of red tide