应用 BP 网络预测水泥土重力式支护结构的水平变形

高沛峻　　邓安福　　王义凤
（重庆建筑大学建筑工程学院，重庆，400045）

提　要　本文首次将人工神经网络应用于水泥土重力式支护结构水平变形的预测中，并通过自编程序证明了该法的可行性，得出的结论对设计和施工均具有重要的指导价值。研究表明，该法不仅可以综合考虑各种因素的影响，而且可以十分有效地处理常规方法难以解决的问题。

当前，水泥土重力式支护结构在深基坑支护中应用十分广泛，其原因是它具有止水和挡土双重作用，且造价低，适用于各种基坑形状，并可在密集建筑群中施工。但是，由于该法在应用时易产生较大的水平变形，对周围环境影响很大。因而在应用时，对基坑变形的预测，就显得十分重要。深基坑开挖过程中，侧壁变形受到工程地质条件、周围环境、施工方法、基坑支护型式等多种因素的综合影响，而其中每一种因素又极其复杂，采用常规方法对基坑变形进行预测十分困难。而人工神经网络则具有解决复杂的、不确定性的、非线性问题的能力，因而在对基坑变形进行预测方面具有明显的优势。

一　BP 网络的结构与数学描述

BP 网络是由非线性变换单元组成的前馈型人工神经网络，它的一个重要特点是由于其变换函数 $f(u)$ 是连续可微的，因而可以严格利用梯度法进行推算，它的权的学习解析式十分明确，其学习算法称为反向传播算法（Back-Propagation），简称 BP 算法，因而这种网络也称为 BP 网络[1]。

BP 网络每个神经元满足下列方程：

$$s_j = \sum w_i x_i - \theta_j \tag{1}$$

$$u_j = s_j \tag{2}$$

$$y_j = f(u_j) \tag{3}$$

式中，s_i—表示神经元 i 突触后电位的累加之值；

θ_i—阈值；

u_i—细胞 i 的状态，可直接简化为 $u_i = s_i$。

y_i—表示神经元 i 的输出。

$f(u)$—输入输出变换函数，可取如下所示的非线性变换：

$$f(u_j)=\frac{1}{1+e^{-u_j}}=\frac{1}{1+e^{-(\sum w_i x_i-\theta_j)}} \tag{4}$$

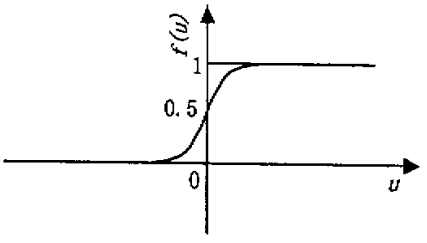


图 1　输入输出非线性函数

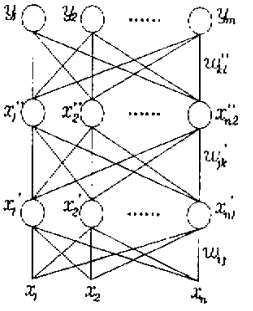


图 2　多层 B-P 网络(四层)

函数 $f(u)$ 的图形如图 1 所示，它是一个连续可微的函数，其一阶导数存在，且满足输入与输出为非线性单调上升的关系。图 2 所示是一个典型的多层 BP 网络的结构。设输入层有 n 个神经元，输入矢量为 $\vec{x}\in R^n$，$\vec{x}=(x_1,x_2,\cdots,x_n)^T$；第二层有 n_1 个神经元 $\vec{x'}\in R^{n_1}$，$x'=(x'_1,x'_2,\cdots,x'_{n_1})^T$；第三层有 n_2 个神经元 $\vec{x''}\in R^{n_2}$，$x''=(x''_1,x''_2,\cdots,x''_{n_2})^T$；最后输出 m 个神经元，$\vec{y}\in R^m$，$\vec{y}=(y_1,y_2,\cdots,y_m)^T$。如输入与第二层之间的连接权为 w_{ij}，阈值为 θ_j；第二层与第三层之间的连接权为 w'_{jk}，阈值为 θ'_k；第三层与最后层之间的连接权为 w''_{kl}，阈值为 θ''_l，则各层神经元的输出满足：

$$\begin{cases} y_l=f(u''_l)=f(\sum_{k=1}^{n_2} w''_{kl}\cdot x''_k-\theta''_l) & \text{最后层输出} \\ x''_k=f(u'_k)=f(\sum_{j=1}^{n_1} w'_{jk}\cdot x'_j-\theta'_k) & \text{第三层输出} \\ x'_j=f(u_j)=f(\sum_{i=1}^{n} w_{ij}\cdot x_i-\theta_j) & \text{第二层输出} \end{cases} \tag{5}$$

上式中，$l=1,2,\cdots,m$；$k=1,2,\cdots,n_2$；$j=1,2,\cdots,n_1$；$i=1,2,\cdots,n$。

二　BP网络的学习算法

BP 网络的基本思想是利用 LMS 算法，通过梯度搜索技术使网络的希望输出和实际输出的均方差最小。在学习时，首先需要提供足够样本的教师，然后根据误差修改权的初始值，直到满足要求为止。如果要输入的学习样本数为 $x^1,x^2,\cdots,x^P$，已知与其对应的教师

为 $t^1,t^2,\cdots,t^P$，LMS 算法是用实际的输出 $y^1,y^2,\cdots,y^P$ 与教师 $t^1,t^2,\cdots,t^P$ 之间的误差来修改其连接权和阈值，使 $\overrightarrow{y^{P_1}}$ 与要求的 $\overrightarrow{t^{P_1}}$ 尽可能接近。为方便起见，计算时常把阈值写入连接权中，令 $\theta''_l = w''_{0l}$，$\theta'_k = w'_{0k}$，$\theta_j = w_{0j}$；$x_0 = x'_0 = x''_0 = -1$。则方程(5)改为：

$$\begin{cases} y_l = f(u''_l) = f(\sum_{k=0}^{n_2} w''_{kl} \cdot x''_k) \\ x''_k = f(u'_k) = f(\sum_{j=0}^{n_1} w'_{jk} \cdot x'_j) \\ x'_j = f(u_j) = f(\sum_{i=0}^{n} w_{ij} \cdot x_i) \end{cases} \tag{6}$$

如果第 P_1 个样本输入到图 2 所示的网络中，得到输出 y_l，$l = 1,2,\cdots,m$。则误差为：

$$E_{P_1} = \frac{1}{2}\sum_{l=1}^{m}(t_l^{P_1} - y_l^{P_1})^2 \tag{7}$$

对于P个学习样本其总误差为：

$$E_{总} = \sum_{P_1=1}^{P} E_{P_1} = \frac{1}{2}\sum_{P_1=1}^{P}\sum_{l=1}^{m}(t_l^{P_1} - y_l^{P_1})^2 \tag{8}$$

采用梯度下降法，对每个 w_{sq} 的修正值为：

$$\Delta w_{sq} = -\sum_{P_1}^{P} \eta \frac{\partial \varepsilon}{\partial w_{sq}} \tag{9}$$

式中 η 为步长。

则由此可推出多层 BP 网络各层之间权修正的基本表达式如下：

$$w''_{kl}(n_0+1) = w''_{kl}(n_0) + \eta\sum_{P_1=1}^{P}\delta_{kl}^{P_1} x''^{P_1}_k \tag{10}$$

$$w'_{jk}(n_0+1) = w'_{jk}(n_0) + \eta\sum_{P_1=1}^{P}\delta_{jk}^{P_1} x'^{P_1}_j \tag{11}$$

$$w_{ij}(n_0+1) = w_{ij}(n_0) + \eta\sum_{P_1=1}^{P}\delta_{ij}^{P_1} x_i^{P_1} \tag{12}$$

当权的修正是在所有样本输入后通过计算其总的误差而进行的，则这种修正被称为批处理。批处理修正可以保证其 $E_{总}$ 向减小的方向变化，在样本数多的时候，它比分别处理时的收敛速度快。

由以上分析可以看出，整个 BP 网络的学习过程分为两个阶段：第一个阶段是从网络的底部向上进行计算，如果网络的结构的权已设定，输入已知学习样本，可按公式(6)计算每一层的神经元输出；第二个阶段是对权值和阈值的修改，这是从最高层向下进行计算和修改。从已知最高层的误差修改与最高层相联的权，然后按公式(10)、(11)、(12)修改各层的权；两个过程反复交替，直到达到收敛为止。具体计算步骤如下：

1. 对图 2 所示网络各层的权和阈值赋初值，可取一个随机数加到各层上，使 $w_{sq}(0) = Random(\cdot)$　　sq 为 ij、jk、kl；

2. 在已知 P 个学习样本中，依次将样本输入到图 2 所示的网络中。先取一个样本 $P_1 = 1$，将其输入 $x_i^{P_1}$ 和期望输出(教师) $t_l^{P_1}$ 输入到网络中；

3. 按公式(6)分别计算各层的输出 x'_j，x''_k，y_l；

4. 计算网络输出误差：第 P_1 个样本的输出误差按公式(7)进行计算；

5. 待所有样本均输入网络后，按公式(8)进行网络总误差计算。如果 $E_{总} < \varepsilon$，则学习停止，其中 ε 为大于 0 的一个给定小数；否则转到步骤(6)；

6. 反向逐层计算网络各层节点误差。对已知样本的教师可得：

$$\delta_{kl}^{P_1} = (t_l^{P_1} - y_l^{P_1}) y_l^{P_1} (1 - y_l^{P_1}) \tag{13}$$

$$\delta_{jk}^{P_1} = \sum_{l=1}^{m} \delta_{kl}^{P_1} w''_{kl}\, x''^{P_1}_{k} (1 - x''^{P_1}_{k}) \tag{14}$$

$$\delta_{ij}^{P_1} = \sum_{k=0}^{n_2} \delta_{jk}^{P_1} w'^{P_1}_{jk}\, x'^{P_1}_{j} (1 - x'^{P_1}_{j}) \tag{15}$$

7. 按公式(10)、(11)、(12)修改各层的权和阈值，回到步骤(2)继续进行计算，直到满足要求为止。

三　水泥土重力式支护结构水平变形预测的神经网络模型

水泥土重力式支护结构水平变形预测的神经网络模型采用三层结构(如图 3 所示)，输入层包括五个单元，分别为时间 x_1(从基坑开挖开始到计算位移时的时间)、基坑开挖深度 x_2、支护结构埋入基坑底以下的深度 x_3、支护结构宽度 x_4、计算点深度 x_5。隐层包括 11 个单元，输出层仅有一个单元 y，即支护结构水平变形。

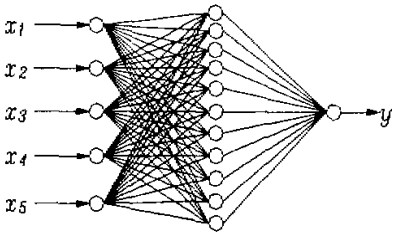


图 3　人工神经网络应用模型

在具体应用时，各输入量不能直接输入到网络中，须先将各输入量进行规一化处理，将它们变化为 0～1 之间的值，然后再输入网络中进行计算。同样，经过网络

计算得到的结果，也必须经还原后，才成为所要预测的支护结构水平变形值。本例中，x_1、x_4和x_5分别以100天、10米和20米为基数进行规一化处理，而x_2和x_3则以二者之和为基数进行规一化处理。输出层单元y以0.1米单位。

四 实例验证

表1 人工神经网络学习样本

输入样本编号	网络输入值					样本教师值t	网络输出值y
	x_1	x_2	x_3	x_4	x_5		
1	0.05	0.536	0.464	0.320	0.0545	0.0940	0.0989
2	0.05	0.536	0.464	0.320	0.1000	0.0933	0.0958
3	0.05	0.536	0.464	0.320	0.1530	0.091	0.0877
4	0.05	0.536	0.464	0.320	0.2000	0.085	0.0917
5	0.05	0.536	0.464	0.320	0.3000	0.069	0.0777
6	0.05	0.536	0.464	0.320	0.4000	0.053	0.0662
7	0.05	0.536	0.464	0.320	0.5000	0.048	0.054
8	0.05	0.536	0.464	0.320	0.6000	0.040	0.0422
9	0.09	0.536	0.464	0.320	0.0545	0.176	0.1821
10	0.09	0.536	0.464	0.320	0.1000	0.168	0.1752
11	0.09	0.536	0.464	0.320	0.2000	0.160	0.1580
12	0.09	0.536	0.464	0.320	0.3000	0.139	0.1381
13	0.09	0.536	0.464	0.320	0.4000	0.115	0.1162
14	0.09	0.536	0.464	0.320	0.5000	0.096	0.0936
15	0.09	0.536	0.464	0.320	0.6000	0.075	0.0722
16	0.15	0.536	0.464	0.320	0.0545	0.304	0.2883
17	0.15	0.536	0.464	0.320	0.1000	0.293	0.2762
18	0.15	0.536	0.464	0.320	0.2000	0.267	0.2471
19	0.15	0.536	0.464	0.320	0.3000	0.227	0.2143
20	0.15	0.536	0.464	0.320	0.4000	0.181	0.1789
21	0.15	0.536	0.464	0.320	0.5000	0.149	0.1429
22	0.15	0.536	0.464	0.320	0.6000	0.112	0.1093
23	0.47	0.536	0.464	0.320	0.0545	0.450	0.4906
24	0.47	0.536	0.464	0.320	0.1000	0.443	0.4699
25	0.47	0.536	0.464	0.320	0.1563	0.448	0.4425
26	0.47	0.536	0.464	0.320	0.2000	0.400	0.4197
27	0.47	0.536	0.464	0.320	0.3000	0.339	0.3628
28	0.47	0.536	0.464	0.320	0.4000	0.299	0.3010
29	0.47	0.536	0.464	0.320	0.5000	0.245	0.2383
30	0.47	0.536	0.464	0.320	0.6000	0.173	0.1801
31	0.61	0.536	0.464	0.320	0.0545	0.573	0.5689
32	0.61	0.536	0.464	0.320	0.1000	0.560	0.5466
33	0.61	0.536	0.464	0.320	0.1563	0.555	0.5165
34	0.61	0.536	0.464	0.320	0.2000	0.507	0.4912
35	0.61	0.536	0.464	0.320	0.3000	0.421	0.4271

本文根据文献[3]提供的实例，选取了40个样本进行验证(所有样本均经作者进行了适当的转化)。其中以35个样本进行学习，5个样本作为测试。学习样本如表1所示，其中

学习误差取 0.0001，步长取 0.01，学习次数共计 1990314 次。测试结果见表 2。

由表 2 可以看出，预测值与实测值的最大误差为 8.9%，完全可以满足工程需要。

表 2　网络预测值与实测值之间的比较

输入样本编号	网络输入值					网络输出值 y	网络预测变形值（mm）	实际变形测量值（mm）	预测值与实测值误差百分比
	x_1	x_2	x_3	x_4	x_5				
1	0.6100	0.5360	0.4640	0.3200	0.367	0.3800	38.00	40.00	5.0%
2	0.6100	0.5360	0.4640	0.3200	0.4000	0.3561	35.61	37.90	6.0%
3	0.6100	0.5360	0.4640	0.3200	0.5000	0.2828	28.28	30.90	8.5%
4	0.6100	0.5360	0.4640	0.3200	0.5563	0.2432	24.32	26.70	8.9%
15	0.6100	0.5360	0.4640	0.3200	0.6000	0.2142	21.42	19.70	8.7%

五　结　论

作者通过自编程序，验证了在水泥土重力式支护结构水平变形预测中应用人工神经网络的可行性。这种尝试的成功，不仅拓宽了人工神经网络的应用范围，也为解决工程中复杂的、模糊的、不确定性问题找到了一条出路，具有重要的理论意义和实践价值。

参考文献

[1] 张立明编著：《人工神经网络的模型及其应用》，复旦大学出版社，1994.8.

[2] 龚晓南主编：《深基坑工程设计施工手册》，中国建筑工业出版社，1998.7.