

# 基于神经网络的老年痴呆症数学模型的研究与数值模拟

张岚 陈强

长春大学 吉林长春 130021

**【摘要】** 神经网络模型可以忽略数据间的关系，而且对于数据充足和数据缺失的两种情况都有较为良好的适应性，因此本文选取神经网络模型，并结合主成分分析法作为研究老年痴呆症数学模型特征方程特征值的方法。

**【关键词】** 老年痴呆症；神经网络模型；数值模拟

## 一、研究背景

对于老年痴呆症现有的生物特征值检测方法是

以  $\beta$ -淀粉样蛋白的浓度作为主要的判别特征。然而随着医学的进步，发现致病原因也不单单是  $\beta$ -淀粉样蛋白，而其形成是一个十分复杂的过程。其反应过程如下图：

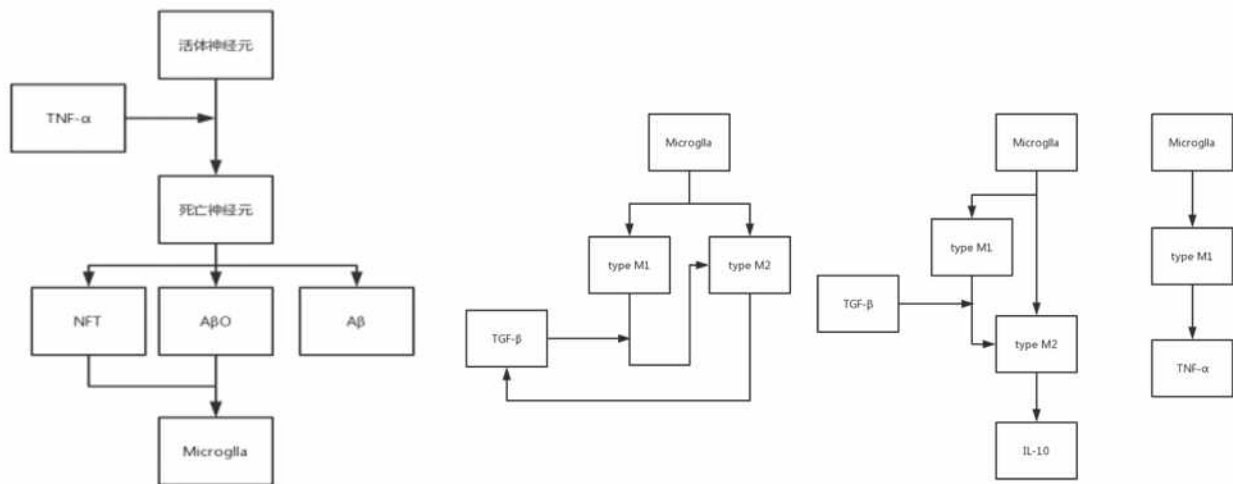


图1 拆解后的实际反应机制

## 二、研究方法

神经网络模型的优势是利用对数据的非线性逼近，使结果的精度取得了理想的提升，因此神经网络开始广泛的应用于医学领域。神经网络的隐藏层，对于非线性数据能够做到有效处理，并在处理后可以对其进行相关精度最高的模型建立，传统上对于非线性数据，神经网络一般采取在隐藏层进行高斯传递，从而对数据进行非线性映射。因此，采用神经网络模型来对老年痴呆患者进行预测。但是影响老年痴呆疾病的因素有很多，全部作为神经网络模型的输入变量会导致计算效率与精度的降低。

人工神经网络对于数据的处理有着较高的容错率，这是因为每一个数据都看作是神经元，输入层、隐含层、输出层之间的关系均为神经系统自行拟合的模型，因此数据之间本身存在的相关性并不影响神经网络最终的精度。

计算阈值采用以下函数：
$$g(net) = \begin{cases} 1, net \geq 0 \\ 0, net < 0 \end{cases}$$

得到阶梯函数，输出  $y$  为 
$$y_x = \begin{cases} 1, net_x \geq 0 \\ 0, net_x < 0 \end{cases}$$

其中，该模型称为神经元 M-P 模型。

S 型函数为激活函数  $g(x)$ ，其形式为：
$$g(x) = \frac{1}{1 + e^{-x}}$$

$$H_j = g\left(\sum_{i=1}^n \omega_{ij} x_i + a_j\right)$$

隐含层输出为

$$net_k = \sum_{i=1}^n \omega_{ki} x_i - \Theta_k$$

输出层为：
$$O_k = \sum_{j=1}^i H_j \omega_k + b_k$$

初始值的设定，无疑是对神经网络学习精度的最大影响因素。但是对于国内的老年痴呆群体，数据的获取成为了研究过程中最大的问题，因此，在没有数据的时候，一方面是无法通过 matlab 进行检验，因此对于模型的正确性抱有疑问，另一方面由于缺少数据，无法避免的造成精确度的降低，所以在另一方面证明了模型的通用性还需要增强，对于特殊情况下的模型依旧适用性较低。

## 三、神经网络模型模拟

由于数据的高度缺失，所以对于老年痴呆特征值的神经网络模型的分析验证就需要从计算结果的准确性转换成具体步骤的可操作性来验证。通过每一步的正确性来验证神经网络模型对于老年痴呆模型的特征值的可操作性。

假设输入层初始神经元有  $n$  个，隐层有  $p$  个，输出层有  $q$  个。

以图 1 为例定义变量：

将每个反应的最上层的反应物作为初始输入量：

$x = (x_1, x_2, \dots, x_n)$ ：

将每个反应第二层作为隐藏初始输入量:  $hi=(hi_1,hi_2,\dots,hi_p)$ ;

下一层的反应物为隐藏初始输出量:  $ho=(ho_1,ho_2,\dots,ho_p)$ ;

输出层的第  $i$  次输入量:  $yi=(yi_1,yi_2,\dots,yi_q)$

输出层的第  $i$  次输出量:  $yo=(yo_1,yo_2,\dots,yo_q)$

输出的期望向量:  $dq=(d_1,d_2,\dots,d_q)$

输入层与隐层之间的连接权值  $W_{ih}$

隐层与输出层之间的连接权值  $W_{ho}$

隐藏层之间的阈值  $b_h$

输出层各神经元阈值  $b_o$

假设样本个数为  $k = 1, 2, \dots, m$ 。

第一步: 对数据进行初始化, 并对所有连接权值赋值, 取值范围为  $(-1, 1)$ , 设定误差为  $e$ , 设定精度值为  $\varepsilon$ , 最大计算次数为  $M$ 。

第二步: 随机选取第  $k$  个输入样本, 并计算对应的期望输出值。

$$x(k) = (x_1(k), x_2(k), \dots, x_n(k))$$

$$d_o(k) = (d_1(k), d_2(k), \dots, d_q(k))$$

第三步: 对于隐藏层的输入和输出值进行计算。

$$hi_h(k) = \sum_{i=1}^n w_{ih} x_i(k) - b_h, h = 1, 2, \dots, p$$

$$ho_h(k) = f(hi_h(k)), h = 1, 2, \dots, p$$

$$yi_o(k) = \sum_{h=1}^p w_{ho} ho_h(k) - b_o, o = 1, 2, \dots, p$$

$$yo_o(k) = f(yi_o(k)), o = 1, 2, \dots, p$$

第四步: 比较期望值与真实值, 并计算误差值, 同时计算误差函数对于输出层每个神经元的偏导数  $\delta_o(k)$

$$\frac{\partial e}{\partial w_{ho}} = \frac{\partial e}{\partial yi_o} \frac{\partial yi_o}{\partial w_{ho}}$$

$$\frac{\partial yi_o(k)}{\partial w_{ho}} = \frac{\partial (\sum_h w_{ho} ho_h(k) - b_o)}{\partial w_{ho}} = ho_h(k)$$

第五步: 以连接权值 (输入与输出函数之间)、输出层的  $\delta_o(k)$ , 和隐藏层的输出值, 计算误差函数对隐层各神经元的偏导数  $\delta_h(k)$ 。

$$\frac{\partial e}{\partial w_{ho}} = \frac{\partial e}{\partial yi_o} \frac{\partial yi_o}{\partial w_{ho}} = -\delta_o(k) ho_h(k)$$

$$\frac{\partial e}{\partial w_{ih}} = \frac{\partial e}{\partial hi_h(k)} \frac{\partial hi_h(k)}{\partial w_{ih}}$$

$$\frac{\partial e}{\partial hi_h(k)} = \frac{\partial (\frac{1}{2} \sum_{o=1}^q (d_o(k) - yo_o(k))^2)}{\partial ho_h(k)} \frac{\partial ho_h(k)}{\partial hi_h(k)} =$$

$$\frac{\partial (\frac{1}{2} \sum_{o=1}^q (d_o(k) - f(\sum_{h=1}^p w_{ho} ho_h(k) - b_o))^2)}{\partial ho_h(k)} \frac{\partial ho_h(k)}{\partial hi_h(k)}$$

$$= -(\sum_{o=1}^q \delta_o(k) w_{ho}) f'(hi_h(k)) - \delta_h(k)$$

第六步: 通过比较不同输出层的不同神经元的  $\delta_o(k)$  与隐含层的输出值, 用来修正连接权值  $W_{ho}(k)$

$$\Delta w_{ho}(k) = -\mu \frac{\partial e}{\partial w_{ho}} = \mu \delta_o(k) ho_h(k)$$

$$w_{ho}^{N+1} = w_{ho}^N + \eta \delta_o(k) ho_h(k)$$

第七步: 比较  $\delta_h(k)$  与连接权值。

$$\Delta w_{ih}(k) = -\mu \frac{\partial e}{\partial hi_h(k)} \frac{\partial hi_h(k)}{\partial w_{ih}} = \delta_h(k) x_i(k)$$

$$w_{ih}^{N+1} = w_{ih}^N + \eta \delta_h(k) x_i(k)$$

第八步: 计算误差

$$E = \frac{1}{2m} \sum_{k=1}^m \sum_{o=1}^q (d_o(k) - y_o(k))^2$$

若计算结果误差函数大于假定精度或者学习次数多于假定的最大次数, 则神经网络计算算法结束; 否则, 重复第三步, 进入下一轮。

#### 四、三层神经网络算法的 matlab 实现

```
clear,clc,close all
% 构造样例数据
x = linspace(-10,10,2000);
y = sin(x);
% 训练测试集分割
a = rand(length(x),1);
[m,n] = sort(a);
x_train = x(n(1:floor(0.7*length(a))));
x_test = x(n(floor(0.7*length(a))+1:end));
y_train = y(n(1:floor(0.7*length(a))));
y_test = y(n(floor(0.7*length(a))+1:end));
% 数据归一化
[x_train_regular,x_train_maxmin] = mapminmax(x_train);
x_train_regular = x_train_regular;
[x_test_regular = mapminmax('apply',x_test,x_train_maxmin);
x_test_regular = x_test_regular;
```

#### 五、结语

相较于发达国家, 我国对于老年痴呆患者的数据库的建立与数据的开放程度都较低, 因此, 在建立模型上无疑会增大难度。由于老年痴呆症患者的数据体量极大, 数据间多少会产生联系, 若不作处理, 无疑会对特征值的产生影响。因此在改进后, 选取神经网络模型作为研究特征值的方法, 神经网络模型可以忽略数据间的关系, 而且对于数据充足和数据缺失的两种情况都有较为良好的适应性, 所以在最终的特征值研究中, 选取神经网络作为最终的计算方式较为合理。

课题: 吉林省教育厅“十三五”科学技术研究规划项目《趋化性数学模型及老年痴呆症患者脑内老年斑数学模型的建立与数值模拟》项目编号: 2018LY508L29

## 参考文献

- [1]Zheng Haiping Hu Zhiping Lu Wei[1].Statins and intracerebral hemorrhage[J]. 中华医学杂志: 英文版 ,2014(13):2531-2536.
- [2]Jan A Coebergh[1], 包靖玲 [2]. 诊断痴呆的困难和危险 [J]. 英国医学杂志: 中文版 (BMJ),2014(4):271-271.
- [3]Qian Liu[1,2],Cheng Xu[1,2,5],Guixiang Ji[3],Hui Liu[1,2],Wentao Shao[1,2],Chunlan Zhang[1,2],Aihua Gu[1,2],Peng Zhao[4].Effect of exposure to ambient PM2.5 pollution on the risk of respiratory tract diseases: a meta-analysis of cohort studies[J]. 生物医学研究杂志: 英文版 ,2017,31(2):130-142.
- [4]Naizhi Wang Yingying Guo Lili Yang Wenyi Fu Yanbing Xu Linxin Hou Shuai Zhao Ning Zhang[1].Effect of tumor necrosis factor inhibitors on rheumatoid arthritis-induced peripheral neuropathy A cohort study[J]. 中国神经再生研究: 英文版 ,2012,7(11):862-866.