Operations Research and Fuzziology 运筹与模糊学, 2022, 12(4), 1110-1118
Published Online November 2022 in Hans. http://www.hanspub.org/journal/orf
https://doi.org/10.12677/orf.2022.124117

Hans汉斯

基于BP神经网络的ERα生物活性定量预测模型

张　翠

上海工程技术大学管理学院，上海

收稿日期：2022年9月20日；录用日期：2022年10月19日；发布日期：2022年10月27日

摘　要

本文对抗乳腺癌候选药物优化问题进行了QSAR建模，使用MATLAB软件进行基于BP神经网络的ERα生物活性定量预测模型构建及预测。构建三层BP神经网络，经过莱文贝格－马夸特法和量化共轭梯度法分别模拟10次，选择最终模型。该模型显示，通过17次迭代后残差收敛，R^2均达到0.75以上，均方误差为0.787，模型拟合较好。随后使用模型对test表中50个化合物进行生物活性预测，并将结果填入对应列。

关键词

BP神经网络，建模，优化

Quantitative Prediction Model of ERα Bioactivity Based on BP Neural Network

Cui Zhang

School of Management, Shanghai University of Engineering Science, Shanghai

Received: Sep. 20th, 2022; accepted: Oct. 19th, 2022; published: Oct. 27th, 2022

Abstract

In this paper, the QSAR model was established for the optimization of candidate drugs for breast cancer, and the quantitative prediction model of ERα biological activity based on BP neural network was constructed and predicted by MATLAB software. The three-layer BP neural network is constructed, and the final model is selected after 10 times simulations by the Levinberg-Maquart method and the quantitative conjugate gradient method. The model shows that after 17 iterations, the residuals converge, R^2 reaches more than 0.75, and the mean square error is 0.787. The model fits well. Subsequently, the model was used to predict the biological activities of 50 compounds in the test table, and the results were filled in the corresponding column.

文章引用: 张翠. 基于 BP 神经网络的 ERα 生物活性定量预测模型[J]. 运筹与模糊学, 2022, 12(4): 1110-1118.
DOI: 10.12677/orf.2022.124117

Keywords

BP Neural Network, Modeling and Optimization

1. 引言

2020 年数据调查得出，我国乳腺癌发病人数约为 42 万，并且这一数据有持续上升的趋势。其中对女性而言，这是发病率高居榜首的恶性肿瘤。作为一种激素依赖性肿瘤，乳腺癌发生、发展与癌细胞上的雌激素受体的表达密切相关[1]。根据病理分型，大约 70%的乳腺癌患者是雌激素受体(ER)阳性，这其中有研究发现，雌激素受体 α 亚型(Estrogen receptors alpha, ERα)在不超过 10%的正常乳腺上皮细胞中表达，但大约在 50%~80%的乳腺肿瘤细胞中表达，ER 的异常表达促进乳腺癌的发生及发展。因此临床中抗激素治疗常被用于 ERα 表达的乳腺癌患者，这一方法是通过调节雌激素受体活性来控制体内雌激素水平。因此，ERα 是乳腺癌发展的关键调节因子是治疗乳腺癌的重要靶标，能够拮抗 ERα 活性的化合物可能是治疗乳腺癌的候选药物。

高昂的研发费用和漫长的周期延缓了新药的开发进展，因此，具有高效率和低成本的化合物活性预测模型是筛选潜在活性化合物的有效方法。成功的模型能够预测具有更好生物活性的新化合物分子，或者指导已有活性化合物的结构优化。一个具有良好的生物活性(此处指抗乳腺癌活性)和在人体内具备良好的药代动力学性质和安全性的化合物可以成为良好的候选药物，这两大特性被通称为 ADMET (Absorption 吸收、Distribution 分布、Metabolism 代谢、Excretion 排泄、Toxicity 毒性)性质。预测和评价 ADMET 药物动力学方法是目前药物设计和筛选中一种非常有效的手段[2]，早期分析能够减少药物开发后期的风险，还能为筛选最优化合物提供强有力的理论依据。其中，ADME 主要指化合物的药代动力学性质，描述了化合物在生物体内的浓度随时间变化的规律，T 主要指化合物可能在人体内产生的毒副作用。合格的候选药物至少要满足化合物活性好、易吸收、易排泄代谢速度适中、对人体无毒性等特征，否则很难成为药物，还需要对其进行 ADMET 性质优化。

基于上述研究背景，本文需研究和解决以下问题：选择不超过 20 个分子描述符变量，构建化合物对 ERα 生物活性的定量预测模型，叙述建模过程。然后使用构建的预测模型，对 50 个化合物进行 IC50 值和对应的 pIC50 值预测。即对样本数为 1974 的化合物分子描述符和 pIC50 进行分析，以根据两者之间的关系，构建预测模型，并对新样本进行 pIC50 的预测。采取 BP 神经网络的方法，对数据进行“训练”及“累积”，尽可能多的将分子描述符和 pIC50 之间的关系进行高度非线性映射，并完成预测工作。为更有效地解决问题，提出以下模型假设：

① 假设分子描述符与生物活性和 ADMET 之间具有一定的线性和非线性关系。

② 忽视本文所选取分子描述符之外的因素对问题产生的影响。

③ 不考虑分子描述符相互之间的影响。

④ 不考虑分子描述符对线性或非线性关系的影响。

2. 理论基础

在机器学习和认知科学中，人工神经网络是借助生物神经网络原理的一系列数据性学习模型，如动物中

枢神经系统，尤其是脑系统。根据 Kolrnogov 定理，1 个 3 层的神经网络能够实现对任意非线性函数进行逼近。因此，BP 神经网络具有很强的非线性映射能力。使用上述选择的分子描述符和 pIC50，使用能以任意精度逼近的 BP 神经网络模型，建立 ERα 生物活性定量预测模型，并进行预测。模型输入数据为 nHBAcc、nHBint6 等 10 个分子描述符数据；输出数据为 pIC50，建立的神经网络结构如图 1 所示。其中 x1~x10 是人工神经元的输入信号，即分子描述符；wij 表示链接强度，即从神经元 j 到神经元 i 链接权值，正值为激活，负值为抑制；求和单元将输入信号进行线性组合；函数为转移函数(Transfer Function)或激活函数(Activation Function)，以控制神经元输出幅度，一般在(0,1)或(−1,1)之间；θi 表示阈值(threshold)或偏置(bias)，两者为相反数。

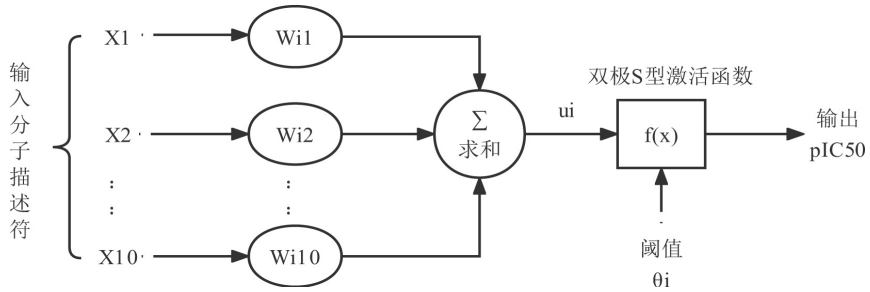


Figure 1. Artificial neural network structure
图 1. 人工神经网络结构

以上可以通过数学表达式展示输入 y 与输出 x 的关系：

$$Y_i = f\left(\sum_{j=1}^{n} w_{ij} x_j - \theta\right) \tag{1}$$

式中， $x1, x2, \cdots, x10$ 表示输入信号；

$wi1, wi2, \cdots, wi10$ 表示神经元 i 的权值；

θ 为阈值；

yi 为神经元 i 的输出。

常用的激活函数有：线性函数、斜坡函数、阈值函数、S (Sigmoid)型和双极 S 型函数。该模型选择 α = 2 的双极 S 型激活函数，即：

$$f(x) = \frac{2}{1+e^{-2x}} - 1, (-1 < f(x) < 1) \tag{2}$$

BP (Back Propagation)神经网络是一种前馈神经网络，即信号向前，误差向后传播。信号传播过程如下图 2 所示，一个神经网络通常包含三层，第一层为输入层，第二层为隐含层，第三层为输出层。

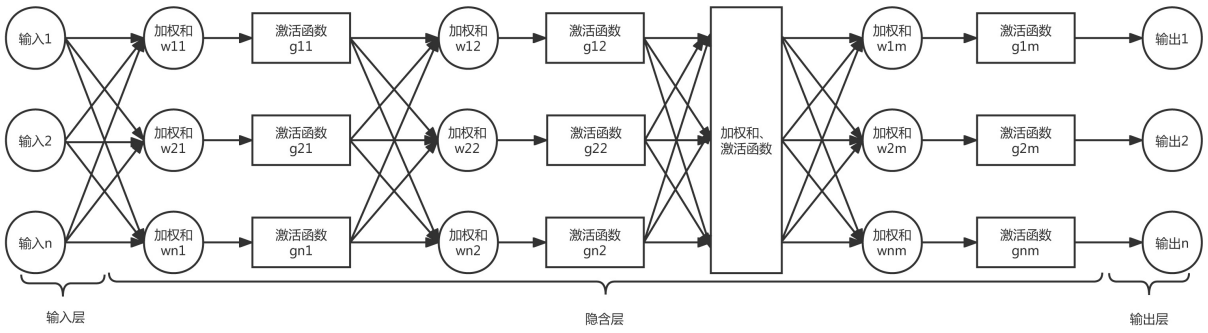


Figure 2. Neural network propagation path
图 2. 神经网络传播路径

3. 研究设计

利用 BP 神经网络建立分子描述符与生物活性的非线性函数关系。具体流程如下图 3 所示。

首先对要参与预测的分子描述符进行标准化处理，然后选取莱文贝格－马夸特方法(Levenberg-Marquardt)和量化共轭梯度法(Scaled Conjugate Gradient)分别尝试 10 次，并在实验前从原始数据中抽出最后 10 个数据来判断拟合度 R^2，找出拟合度较高且均方误差较少的那次实验的数据作为本次的预测模型。

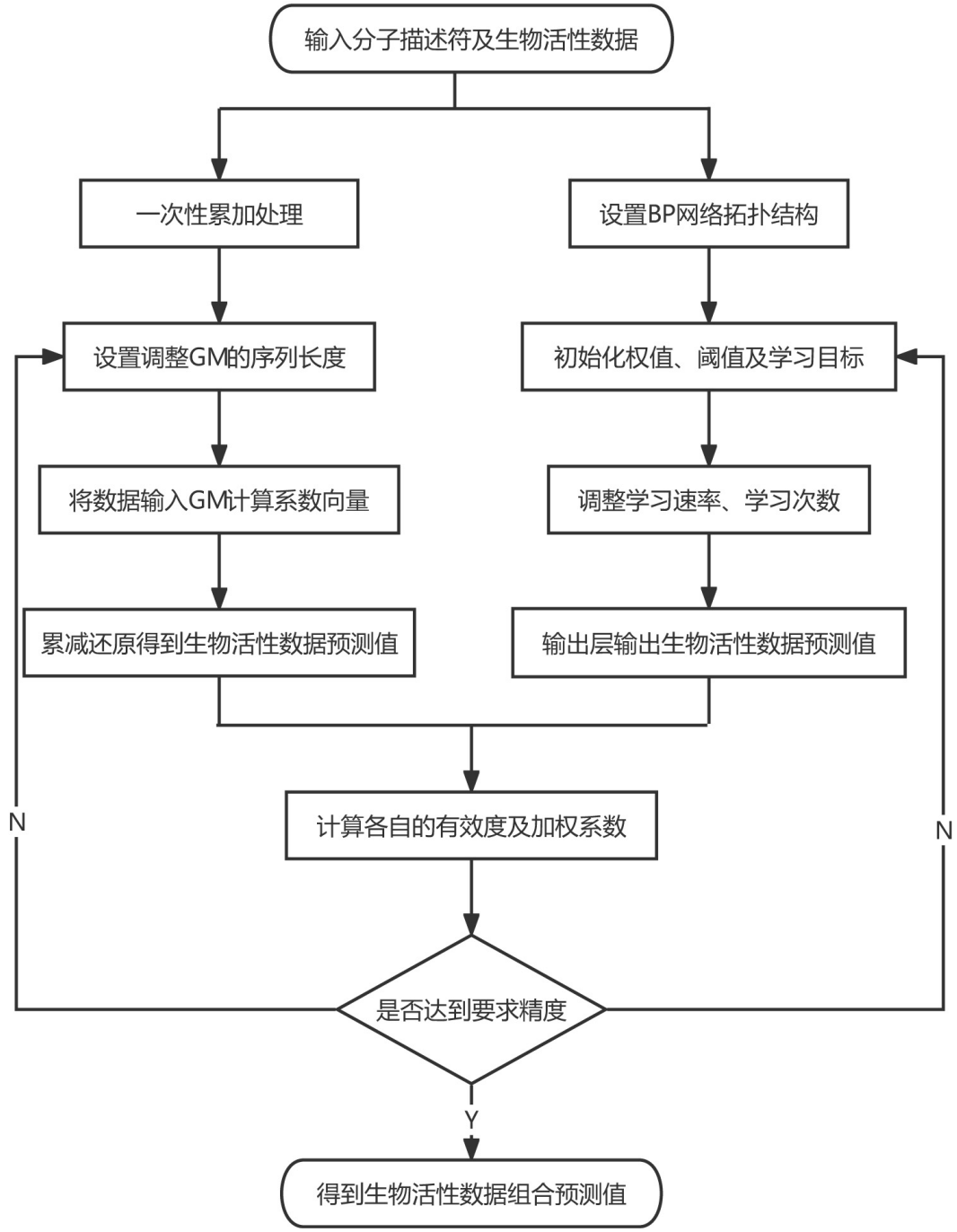


Figure 3. BP neural network operation flow chart
图 3. BP 神经网络操作流程图

通过误差对比，确定最佳的隐含层神经元个数，拟合误差运行结果如表 1 所示。

图 4 表明，在经过 17 次训练后，隐含层神经元为 11 个的 BP 网络对函数的逼近效果最好，误差最小。因此，预测模型选择神经元个数为 11。

Table 1. Network error
表 1. 网络误差

神经元个数	1	2	3	4	…	1962	1963	1964
网络误差	0.010	0.573	0.545	0.839	…	-1.037	0.180	-1.325

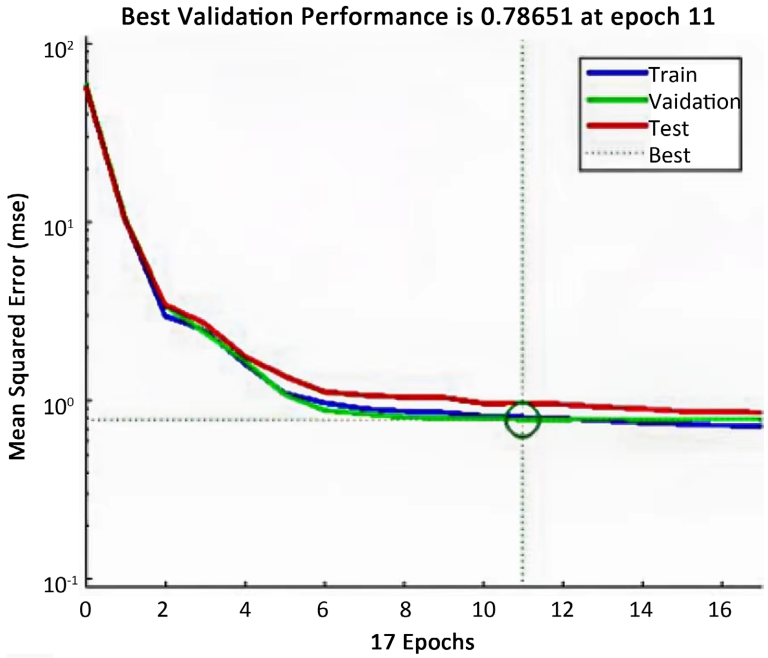
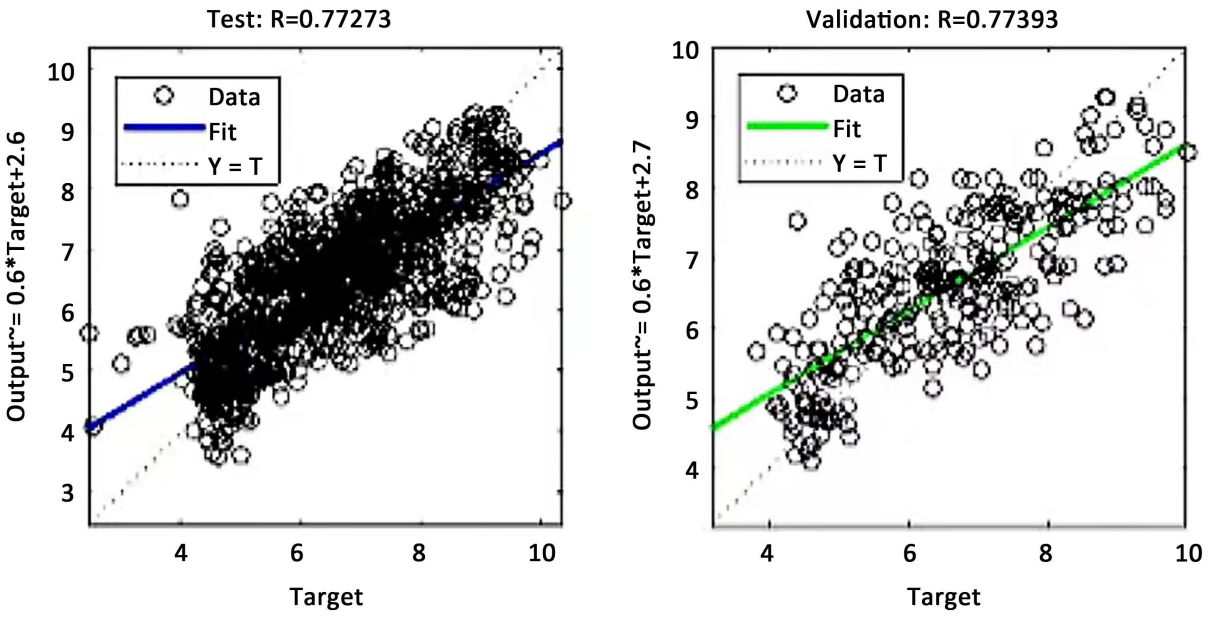


Figure 4. Residual convergence
图 4. 残差收敛



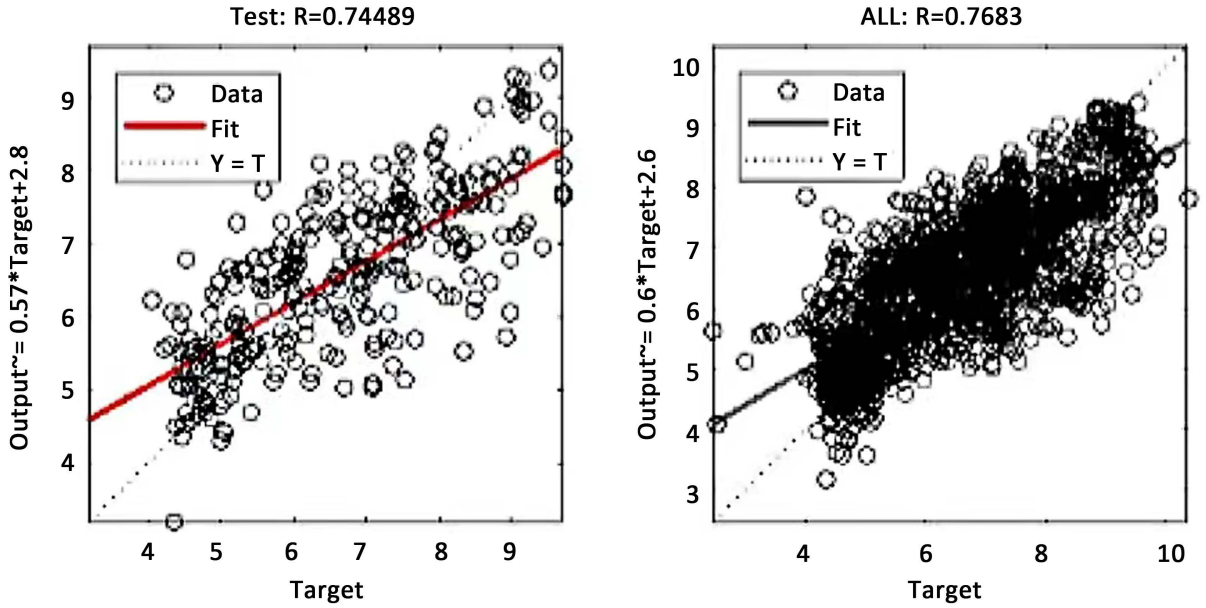


Figure 5. Regression
图 5. Regression 图

通过 10 次模拟，对照选取的 10 个样本所对应的生物活性的预测值与真值的差异性，找出拟合度最高的训练方法和结果，最终得到的拟合结果如图 5 所示，R^2均在 0.7 以上，拟合度较高，结果具有一定的可信度。由于 pIC50 是 IC50 的负对数转化来的，通过推导，其运算公式为：

$$IC50 = 10^{9-PIC50} \tag{3}$$

使用上述构建的预测模型，对文件“ERα_activity.xlsx”的 test 表中的 50 个化合物进行 IC50 值和对应的 pIC50 值进行预测后，同时将结果填入了“ERα_activity.xlsx”的 test 表中的 IC50_nM 列及对应的 pIC50 列，部分结果展示如表 2 所示。

Table 2. Predicted bioactive IC50 and pIC50 results
表 2. 预测的生物活性 IC50 和 pIC50 结果

SMILES	IC50_nM	pIC50
COc1cc(OC)cc(\C=C\c2ccc(OS(=O)(=O)[C@@H]3C[C@@H]4O[C@H]3C(=C4c5ccc(O)cc5)c6ccc(O)cc6)cc2)c1	595.799	6.225
OC(=O)\C=C\c1ccc(cc1)C2=C(CCOc3ccccc23)c4ccc(O)cc4	141.026	6.851
COc1ccc2C(=C(CCOc2c1)c3ccc(O)cc3)c4ccc(\C=C\C(=O)O)cc4	143.087	6.844
……	……	……
CC(C)C[C@H](NC(=O)[C@]1(C)CCC\C=C/C[C@H](C)C[C@](C)(NC(=O)[C@H](CCCCN)NC(=O)[C@H](Cc2cnc[nH]2)NC(=O)C)C(=O)N[C@@H](CC(C)C)C(=O)N[C@@H](Cc3cnc[nH]3)C(=O)N[C@@H](CCCNC(=N)N)C(=O)N1)C(=O)N[C@@H](CCC(=O)N)C(=O)N[C@@H](CC(=O)O)C(=O)N[C@@H](CO)C(=O)N	0.111	9.953
CC\C(=C(/c1ccc(O)cc1)\c2ccc(OCCN(C)C)cc2)\c3ccc(cc3)C(=O)NO	102.188	6.991
CC\C(=C(/c1ccc(O)cc1)\c2ccc(OCCN(C)C)cc2)\c3ccc(CCCCC(=O)NO)cc3	115.744	6.937

4. 模型总结与评价

模型的优点在于：基于 MATLAB 模型的神经网络模型可以模拟多变量而不需要考虑对象复杂的不

确定性和时变性，容易建立预测模型，从而较为精确的预测变量之间的映射关系[3]。

模型的缺点在于：训练集、验证集、测试集的比例 Matlab 会后台自动选择比例来随机抽取样本，因此每次运行的结果可能都不相同。

参考文献

[1] 李晨. “新钥匙”打开乳腺癌耐药“锁” [N]. 中国科学报 2021-09-10(003).

[2] Hodgson, J. (2001) ADMET—Turning Chemicals into Drugs. *Nature Biotechnology*, **19**, 722-726. https://doi.org/10.1038/90761

[3] 郭涤, 周军. 基于 Matlab 的神经网络预测模型研究[J]. 物流科技, 2006, 29(1): 125-128.

附　录

运用 Matlab 中的神经网络求解，由于 IW1_1 输出数据过多，故仅显示前五个作为代表，其代码如下：

```
function [Y,Xf,Af] = myNeuralNetworkFunction(X,~,~)
%输入 1
x1_step1.xoffset =
[-0.959;-0.28044;-2.13884;-6.21238;-4.95147;-0.86629;-0.9518;-0.81305;-0.3341;-2.3047];
x1_step1.gain =
[0.0813653761033145;0.0653666119112348;0.353827618722435;0.107571703263457;0.313105663768352;0.393439011051702;0.110856141984547;0.205235559112972;0.185961574759807;0.497050008201325];
x1_step1.ymin = -1;
% 层次 1
b1 =
[-1.8111078060248325;1.4282247551582381;-0.90375982196551596;-0.43482491375879534;-0.14148385537068167;-0.36312851996008744;-0.72403667783393577;-1.1031963793364412;1.4281464859036888;1.7488087416098097];
IW1_1 = [0.76888607300302514 0.057461449570589615 0.22616072432086912
0.2389943344597017-0.45858207366829273 ……];
% 层次 2
b2 = -0.56306330953903883;
LW2_1 = [-0.6941891684317768 0.20237767688952646 0.85292026637443752 0.22063298452340385
-0.10509355479025818 0.63502770960891264 0.63730818276650569 -0.30593569017895728
0.10969211004056961 -0.012179211610543887];
% 输出 1
y1_step1.ymin = -1;
y1_step1.gain = 0.253774901662226;
y1_step1.xoffset = 2.456;
% 格式输入参数
isCellX = iscell(X);
if ~isCellX
X = {X};
end
% 求解矩阵大小
TS = size(X,2);
if ~isempty(X)
Q = size(X{1},1);
Q = 0;
end
%分配输出
```

```
Y = cell(1,TS);
% 循环
for ts=1:TS
% 输入 1
X{1,ts} = X{1,ts}';
Xp1 = mapminmax_apply(X{1,ts},x1_step1);
% 层次 1
a1 = tansig_apply(repmat(b1,1,Q) + IW1_1*Xp1);
% 层次 2
a2 = repmat(b2,1,Q) + LW2_1*a1;
% 输出 1
Y{1,ts} = mapminmax_reverse(a2,y1_step1)Y{1,ts} = Y{1,ts}';
end
% 最后的延迟状态
Xf = cell(1,0);
Af = cell(2,0);
% 格式输入参数
if ~isCellX
Y = cell2mat(Y);
end
end
%映射最小和最大输入处理功能
function y = mapminmax_apply(x,settings)
y = bsxfun(@minus,x,settings.xoffset);
y = bsxfun(@times,y,settings.gain);
y = bsxfun(@plus,y,settings.ymin);
end
% 对称传递函数
function a = tansig_apply(n,~)
a = 2 ./ (1 + exp(-2*n)) - 1;
end
% 映射最小和最大输出的逆向处理功能
function x = mapminmax_reverse(y,settings)
x = bsxfun(@minus,y,settings.ymin);
x = bsxfun(@rdivide,x,settings.gain);
x = bsxfun(@plus,x,settings.xoffset);
end
```