第 16 卷 第 2 期
2004 年 6 月

江 苏 工 业 学 院 学 报
JOURNAL OF JIANGSU POLYTECHNIC UNIVERSITY

Vol. 16 No. 2
Jun. 2004

文章编号: 1005 - 8893 (2004) 02 - 0054 - 03

二维神经元模型研究*

闫玉宝
(江苏工业学院 计算机工程与科学系, 江苏 常州 213016)

摘要: 神经元是神经网络的中枢, 经过对一维神经元结构的分析, 针对神经元对输入信号的不同整合过程, 提出了采用二元函数作为传输函数来建立二维神经元模型的思想方法, 介绍了单输入二维神经元和多输入二维神经元的结构。论证了二维神经元感知机的结构设计、学习规则和敛散性, 并给出了学习算法。对于算法的约束条件给出了说明, 利用此算法设计单层二维神经元网络, 解决了线性不可分的异或问题。对于传输函数的设计及其对算法稳定性的影响需要从理论上进一步论证。

关键词: 二维神经元; 感知机; 学习规则
中图分类号: TP 1213　　**文献标识码**: A

单输入神经元模型中累加器输出称为净输入, 它被送入一个传输函数 f, 在 f 中产生神经元的标量输出 a。传输函数模拟了细胞体进行阈值处理及信号的输出功能。对于多输入神经元模型与单输入模型的唯一差别在于多个标量的输入。因此, 这 2 种神经元模型的功能由输入、累加器 (整合)、传输函数 3 部分完全决定。由于对输入信号进行整合采用了累加的形式, 对应的传输函数只能选取一元函数来进行阈值处理。把传输函数是一元函数的神经元称为一维神经元。本文针对神经元对输入信号的整合过程提出了二维整合方式, 采用二元函数作为传输函数来建立神经元模型, 利用建立的神经元模型实现了单层网络识别异或问题。

1　二维神经元模型

1.1　单输入二维神经元

如图 1 所示, 是单输入二维神经元。单输入标量 p 乘上 X 方向的权值 w_1 得到 $w_1 p$, 并将其送入累加器。另一个输入 1 乘上偏量值 b_1, 再将其送入累加器。得到 X 方向的净输入 $X = w_1 p + b_1$; 单输入标量 p 乘上 Y 方向的权值 w_2 得到 $w_2 p$, 并将其送入累加器。另一个输入 1 乘上偏量值 b_2, 再将其送入累加器。得到 Y 方向的净输入 $Y = w_2 p + b_2$; X, Y 被送入一个二元传输函数 $f(X, Y)$, 在 f 中产生神经元的标量输出 a, 因此, $a = f(w_1 p + b_1, w_2 p + b_2)$。

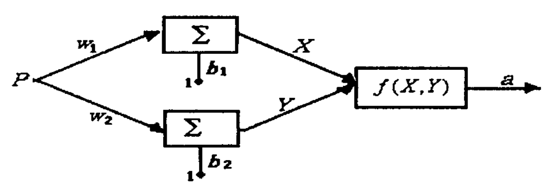

图 1　单输入二维神经元

1.2　多输入二维神经元

如图 2 所示, 是多输入二维神经元。输入向量

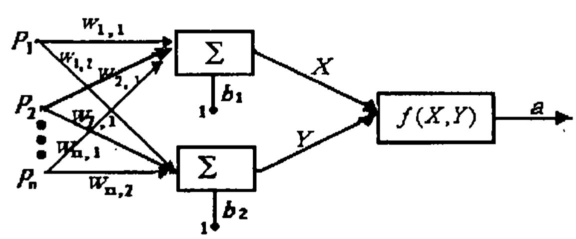

图 2　多输入二维神经元

---

* **收稿日期**: 2004 - 02 - 19
基金项目: 江苏工业学院科技基金资助
作者简介: 闫玉宝 (1965 -), 男, 河南南阳人, 硕士, 主要从事图形图象、CAD 和人工智能的教学与研究工作。

$\mathrm{P}=[p_1, p_2, \cdots, p_m]^{\mathrm{T}}$，$X$ 方向的权向量 $\mathrm{W}_1=[w_{1,1}, w_{2,1}, \cdots, w_{m,1}]^{\mathrm{T}}$，$Y$ 方向的权向量 $\mathrm{W}_2=[w_{1,2}, w_{2,2}, \cdots, w_{m,2}]^{\mathrm{T}}$，$X$ 方向的净输入 $X=\mathrm{W}_1^{\mathrm{T}}\mathrm{P}+b_1$，$Y$ 方向的净输入 $Y=\mathrm{W}_2^{\mathrm{T}}\mathrm{P}+b_2$，神经元的输出 $a=f(\mathrm{W}_1^{\mathrm{T}}\mathrm{P}+b_1, \mathrm{W}_2^{\mathrm{T}}\mathrm{P}+b_2)$。

2 二维神经元感知机的设计

2.1 单输出感知机的结构

如图 3 所示，$f(X, Y)=\begin{cases}1, & XY>0\\ 0, & XY<0\end{cases}$；不妨称 f 为 hardlim2，于是，$a=\text{hardlim2}(X, Y)$。

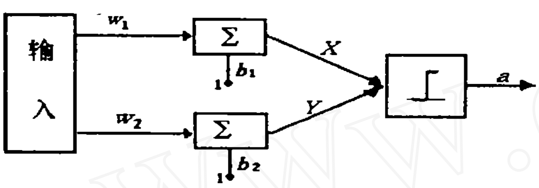

图 3 单输出感知机

判定边界：判定边界由那些使得净输入 X，Y 均为零的输入确定。$\mathrm{W}_i^{\mathrm{T}}\mathrm{P}+b_i=0$ $(i=1, 2)$。判定边界总与权值向量垂直。

2.2 解决异或问题的网络设计

XOR 门的输入/目标对是 $\left\{\mathrm{p}_1=\begin{bmatrix}0\\0\end{bmatrix}, t_1=0\right\}$，$\left\{\mathrm{p}_2=\begin{bmatrix}0\\1\end{bmatrix}, t_2=1\right\}$，$\left\{\mathrm{p}_3=\begin{bmatrix}1\\0\end{bmatrix}, t_3=0\right\}$，$\left\{\mathrm{p}_4=\begin{bmatrix}1\\1\end{bmatrix}, t_4=0\right\}$，对应的是 2 个输入 1 个输出的单神经元感知机（图 4）。

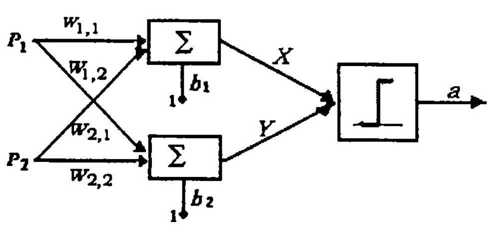

图 4 解决异或问题

由图解（图 5）很容易得到判定边界，如取边界 $p_1=0.5$，$p_2=0.5$。由于 W_1 与 $p_1=0.5$ 垂直，W_2 与 $p_2=0.5$ 垂直，取 W_1，W_2 的方向指向输出为 1 的区域，于是选取权值向量为 $\mathrm{W}_1=\begin{bmatrix}-2\\0\end{bmatrix}$，$\mathrm{W}_2=\begin{bmatrix}0\\2\end{bmatrix}$，权值向量的长度并不十分重要，重要的是他们的方向[1]。

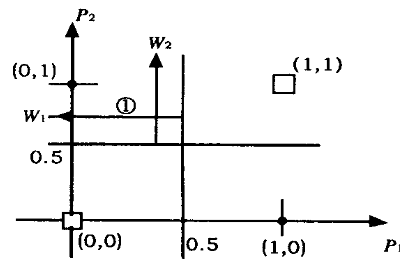

图 5 判定边界图解

由 $b_1=-\mathrm{W}_1^{\mathrm{T}}\mathrm{P}$，$b_2=-\mathrm{W}_2^{\mathrm{T}}\mathrm{P}$，以及 $\mathrm{P}=\begin{bmatrix}0.5\\0.5\end{bmatrix}$，得到 $b_1=1$，$b_2=-1$，于是，$\mathrm{W}=\begin{bmatrix}\mathrm{W}_1^{\mathrm{T}}\\ \mathrm{W}_2^{\mathrm{T}}\end{bmatrix}=\begin{bmatrix}-2 & 0\\ 0 & 2\end{bmatrix}$，$\mathrm{b}=\begin{bmatrix}b_1\\b_2\end{bmatrix}=\begin{bmatrix}1\\-1\end{bmatrix}$，$\begin{bmatrix}X\\Y\end{bmatrix}=\mathrm{WP}+b$。

把 p_1，p_2，p_3，p_4 送入网络，均得到正确识别。

$a=\text{hardlim2}(\mathrm{W}_1^{\mathrm{T}}\mathrm{p}_i+b_1, \mathrm{W}_2^{\mathrm{T}}\mathrm{p}_i+b_2)=t_i$ $(i=1, 2, 3, 4)$

3 学习规则及收敛性

性能指数：$F(x)=E[e^2]=E[(t-a)^2]$，其中 t 为目标输出，a 为实际输出，$e=(t-a)$ 为误差。用 $\hat{F}(x)$ 近似计算均方误差，即均方误差的期望值被第 k 次迭代时的均方误差所代替。

$$\hat{F}(x)=(t(k)-a(k))^2 \tag{1}$$

近似均方误差的最速下降算法为：

$$w_{i,j}(k+1)=w_{i,j}(k)-\alpha\frac{\partial\hat{F}}{\partial w_{i,j}}$$
$$b_i(k+1)=b_i(k)-\alpha\frac{\partial\hat{F}}{\partial b_i} \tag{2}$$

(2) 式中 α 是学习速度。

关于 $\alpha\frac{\partial\hat{F}}{\partial w_{i,j}}$，$\alpha\frac{\partial\hat{F}}{\partial b_i}$ 的计算：

由于 $a(k)=\text{hardlim2}(X, Y)$，$\text{hardlim2}(X, Y)=\begin{cases}1, & XY\geq 0\\ 0, & XY<0\end{cases}$，函数从 X、Y 2 个方向制约函数值的输出，函数值的输出由 XY 决定，X 与 Y 之间是反比例关系。因此，令

$$\frac{\mathrm{d}a(k)}{\mathrm{d}w_{i,j}}=Y\frac{\mathrm{d}X}{\mathrm{d}w_{i,j}}+X\frac{\mathrm{d}Y}{\mathrm{d}w_{i,j}}$$
$$\frac{\mathrm{d}a(k)}{\mathrm{d}b_i}=Y\frac{\mathrm{d}X}{\mathrm{d}b_i}+X\frac{\mathrm{d}Y}{\mathrm{d}b_i} \tag{3}$$

由 (1) 式、(3) 式得，

$$\frac{\partial \hat{F}}{\partial w_{i,j}} = 2\ (a\ (k)\ -t\ (k))\ \frac{d\,a\ (k)}{d\,w_{i,j}} = 2\ (a\ (k)\ -t\ (k))\left(Y\frac{d\,X}{d\,w_{i,j}} + X\frac{d\,Y}{d\,w_{i,j}}\right) \quad (4)$$

$$\frac{\partial \hat{F}}{\partial b_i} = 2\ (a\ (k)\ -t\ (k))\ \frac{d\,a\ (k)}{d\,b_i} = 2\ (a\ (k)\ -t\ (k))\left(Y\frac{d\,X}{d\,b_i} + X\frac{d\,Y}{d\,b_i}\right) \quad (5)$$

由 (2) 式、(4) 式及 (5) 式得，

$$w_{i,j}(k+1) = w_{i,j}(k) + 2\alpha e(k)\left(Y\frac{d\,X}{d\,w_{i,j}} + X\frac{d\,Y}{d\,w_{i,j}}\right)$$

$$b_i(k+1) = b_i(k) + 2\alpha e(k)\left(Y\frac{d\,X}{d\,b_i} + X\frac{d\,Y}{d\,b_i}\right) \quad (*)$$

其收敛性的分析同 LMS 算法。对于一维神经元单层感知机，如果输入样本线性可分，无论感知器的初始权值如何取，经过有限次调整后，总能够稳定到一个权向量，该权向量确定的超平面能把各类样本正确分开[2]。但对于二维神经元单层感知机，当所有权值和偏置值全部初始值赋为 0 时，由 (*) 式进行迭代不能改变它们的值，导致这些参数的值恒为 0。这也说明零输入的反馈信息得不到传递。因此初始值不能全赋为 0，否则得不到解。

下面是异或识别网络的学习过程的迭代公式和实验结果 (在 Microsoft C 编译器上)：

$$w_{1,1}(k+1) = w_{1,1}(k) + 2\alpha e(k)\,(w_{1,2}(k)\,p_1 + w_{2,2}(k)\,p_2 + b_2(k))\,p_1$$

$$w_{1,2}(k+1) = w_{1,2}(k) + 2\alpha e(k)\,(w_{1,1}(k)\,p_1 + w_{2,1}(k)\,p_2 + b_1(k))\,p_2$$

$$w_{2,1}(k+1) = w_{2,1}(k) + 2\alpha e(k)\,(w_{1,2}(k)\,p_1 + w_{2,2}(k)\,p_2 + b_2(k))\,p_1$$

$$w_{2,2}(k+1) = w_{2,2}(k) + 2\alpha e(k)\,(w_{1,1}(k)\,p_1 + w_{2,1}(k)\,p_2 + b_1(k))\,p_2$$

$$b_1(k+1) = b_1(k) + 2\alpha e(k)\,(w_{1,2}(k)\,p_1 + w_{2,2}(k)\,p_2 + b_1(k))$$

$$b_2(k+1) = b_2(k) + 2\alpha e(k)\,(w_{1,1}(k)\,p_1 + w_{2,1}(k)\,p_2 + b_1(k))$$

例如：当时 $W\ (0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$，$b\ (0) = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ 迭代次数 16 次，就可完全识别异或问题。

算法的收敛性基于下面几点：①解存在；②不能零初始化；③解决线性问题，即线性可分及线性不可分问题；④不能解决非线性问题。

4 结 语

本文提出的二维神经元模型，可用来设计单层网络解决异或问题，为设计单层网络解决线性不可分问题提供了示例。对于学习规则算法中零初始化问题还需要从理论上进一步论证。由于二维神经元的结构较为复杂，这给实际应用带来了困难，并且很多问题需要解决，如传输函数如何选取，多层网络如何建立等等，都需要作进一步的研究。

参考文献：

[1] [美] Martim T ·Hagan. 神经网络设计 [M] . 北京：机械工业出版社，2002. 34 - 50.

[2] 韩力群 . 人工神经网络理论、设计及应用 [M] . 北京：化学工业出版社，2002. 35 - 38.

Research of 2 - D Neuron Model

YAN Yu - bao

(Department of Computer Science and Technology , Jiangsu Polytechnic University , Changzhou 213016 , China)

Abstract : Neuron is ganglia of neural network. The constitution of 1 - D Neuron is concisely described in this paper. According to processing of input , the modeling of 2 - D Neuron was introduced. The structure of single - input 2 - D Neuron and multiple - input 2 - D Neuron were introduced. The Perceptron consisted of 2 - D Neuron was devised. Algorithm of learning rule and convergence are given. The Linear unseparated problem of Pattern classification of XOR was resolved. Choice of transfer function and transfer function affecting the stability of the algorithm will be researched further.

Key words : 2 - D neuron ; perceptron ; learning rule