普通高等教育“十五”国家级规划教材

神经网络计算

吴 微 编著

高等教育出版社
HIGHER EDUCATION PRESS

普通高等教育“十五”国家级规划教材

神经网络计算

吴 微 编著

高等教育出版社

内容提要

本书是普通高等教育“十五”国家级规划教材.

本书简要介绍了几种常用的人工神经网络的原理、计算方法和应用，包括以 BP 网络为代表的前馈网络，以 Hopfield 网络为代表的联想记忆网络，径向基函数网络，Boltzmann 机，特征映射网络（SOFM 网络与 ART 网络），以及小脑模型网络等，每章后附有练习题. 全书内容剪裁适当，叙述清晰简明.

本书可作为理工科相关专业的高年级本科生选修课教材和研究生教材，也可作为人工神经网络研究与应用方面的参考书.

图书在版编目(CIP)数据

神经网络计算/吴微编著. —北京:高等教育出版社，
2003.7(2004 重印)
ISBN 7-04-011917-X

Ⅰ.神… Ⅱ.吴… Ⅲ.人工神经元网络-计算方法 Ⅳ.TP183

中国版本图书馆 CIP 数据核字(2003)第 037492 号

出版发行	高等教育出版社	购书热线	010-64054588
社　　址	北京市西城区德外大街 4 号	免费咨询	800-810-0598
邮政编码	100011	网　　址	http://www.hep.edu.cn
总　　机	010-82028899		http://www.hep.com.cn
经　　销	新华书店北京发行所		
印　　刷	北京人卫印刷厂		
开　　本	787×960　1/16	版　　次	2003 年 7 月第 1 版
印　　张	5.5	印　　次	2004 年 5 月第 2 次印刷
字　　数	95 000	定　　价	8.00 元

本书如有缺页、倒页、脱页等质量问题，请到所购图书销售部门联系调换。

策　　划　王　瑜
编　　辑　王　瑜
封面设计　王凌波
责任绘图　朱　静
版式设计　王艳红
责任校对　刘　莉
责任印制　宋克学

致　　谢

感谢国家自然科学基金、国防科工委国防基础科研基金、教育部青年骨干教师基金、辽宁省中青年学科带头人基金和大连理工大学的资助；感谢作者的研究生和博士后（马玉梅、李正学、邵郅邛、侯利昌、郭力宾、张立庆、张玉林、杨洁、南东、郑高峰、邵红梅、李峰、范修宇、张凌）的多方协助；感谢家人在本书写作过程中的鼓励与支持。

目　录

第一章 前传网络

§1.1 引 言

图 1.1 给出了一个简单的单层前传网络(神经元)的示意图. 它也是许多更复杂的神经网络的基本构件之一. 神经元对外界传入的 N 个信号经权值 W 处理后,用线性求和器得到"综合印象",再由活化函数 $g(\cdot)$对此综合印象作出非线性反应 ζ. 这种反应机制是对真正的生物神经元反应机制的一种简单而又常常有效的模拟. 将大量简单神经元按某种方式连接起来,并通过某种学习过程确定单元之间的连接强度(权值 W),就得到各种人工神经网络,用来完成逼近、分类、控制和模拟等各种任务.

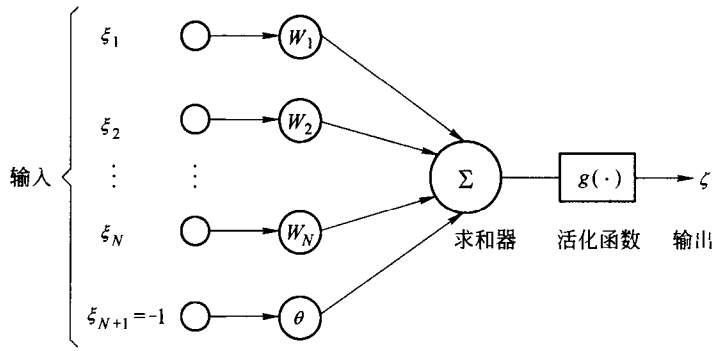

图 1.1 神经元模型

设给定 J 个输入样本模式$\{\xi^j\}_{j=1}^J$,其中 $\xi^j=(\xi_1^j,\cdots,\xi_N^j)^{\mathrm{T}}\in\mathbf{R}^N$,以及理想输出$\{O^j\}_{j=1}^J\subset\mathbf{R}^1$. 另外,给定一个非线性函数 $g(x):\mathbf{R}^1\to\mathbf{R}^1$. 一个单层前传网络(神经元)的任务是选择权向量 $W=(W_1,\cdots,W_N)^{\mathrm{T}}\in\mathbf{R}^N$ 和阈值$\theta\in\mathbf{R}^1$,使得

$$O^j=\zeta^j\equiv g(W\cdot\xi^j-\theta)=g\Big(\sum_{n=1}^{N}W_n\xi_n^j-\theta\Big),\ j=1,\cdots,J \qquad (1.1)$$

其中 ζ^j 为网络的实际输出. 利用样本模式,通过某种学习算法选定 W 之后,我们就可以向网络输入 $\mathbf{R}^N$ 中其他模式向量,得到相应的输出,从而完成各种分类

或逼近任务.

上述函数 $g(x)$称为活化函数,常见的有符号函数、径向基函数、随机值函数等等. 网络的输出值 ζ^j 及理想输出 O^j 可以只取有限个离散值(例如双极值 ± 1 或二进制 0,1),这时网络相当于一个分类器;也可以取连续值,这时网络相当于输入 ξ 与输出 O 之间函数关系的一种数值逼近器. 当存在 W 和 θ 使式(1.1)成立时,我们说该问题是可解的,或样本模式 $\{\xi^j\}_{j=1}^J$ 是可分的;否则,称为不可解的,或不可分的,这时只能选取 W 和 θ 使得误差 $O^j - \zeta^j$ 尽可能地小.

注 1.1 令 $\widetilde{W}=(W_1,\cdots,W_N,\theta)^{\mathrm{T}}$, $\tilde{\xi}^j=(\xi_1^j,\cdots,\xi_N^j,-1)^{\mathrm{T}}$,将 θ 和 W 一起作为新的权值 $\widetilde{W}$ 来进行选择,于是式(1.1)中 ζ^j 的定义可以相应地改为

$$\zeta^j = g(\widetilde{W}\cdot\tilde{\xi}^j) \tag{1.2}$$

这样做可以大大简化记号.

在下一节,我们讨论最简单的神经网络,即取 $g(x)$为符号函数的所谓线性感知器(Adline). §1.3 研究多个线性感知器组成的 Madline 网络. §1.4 讨论最重要的多层前传网络,即 BP 网络. 前传网络的一些简单应用在 §1.5 中给出.

§1.2 自适应线性(Adaptive Linear)感知器

在式(1.1)中,取 $g(x)$为符号函数(见图 2.1):

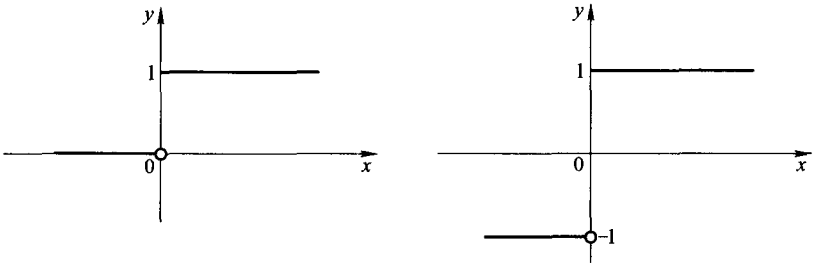

图 2.1 符号函数(函数值为{0,1}或{1,-1})

$$\mathrm{sgn}(x)=\begin{cases}1, & x\geqslant 0\\ -1, & x<0\end{cases} \tag{2.1}$$

且理想输出 O^j 取值亦为 ± 1(也可以考虑符号函数的取值为 0,1. 一般地说,取值为 0,1 时电路实现方便,而取 ± 1 时数学处理比较简单.). 对输入样本模式 ξ^j,网络实际输出为

$$\zeta^j = \mathrm{sgn}(W \cdot \xi^j - \theta) \tag{2.2}$$

网络式(2.2)称为**线性感知器**.

注 2.1 式(2.2)中主要的运算为便于并行处理的向量乘法,而符号函数(以及以后用到的符号函数的各种逼近)则容错性较好. 事实上,神经网络用到的主要运算就是向量乘法,并且广泛采用符号函数及其各种逼近. 神经网络可以用计算机模拟实现. 更重要的是,神经网络还可以用电路、光路等硬件来实现(参见[MT][DK]);这时不论 N 多大,式(2.2)中的向量乘法所需的时间基本不变(参看图 1.1),使得便于并行处理的特点更加突出. 并行、容错、可以硬件实现以及后面将要讨论的自我学习特性,是神经网络的几个基本优点,也是神经网络计算方法与传统计算方法的区别所在.

以 $N=2$ 为例. 线性感知器的目标就是求法向量 W 和阈值 θ,使得与 W 垂直的直线(一般是 $N-1$ 维超平面) $W\cdot\xi=\theta$ 将样本模式 $\{\xi^j\}_{j=1}^J$ 分成 $W\cdot\xi^j>\theta$ 和 $W\cdot\xi^j<\theta$(即 $\zeta^j=1$ 和 $\zeta^j=-1$)两类,分别位于 $W\cdot\xi=\theta$ 的两侧(见图 2.2).

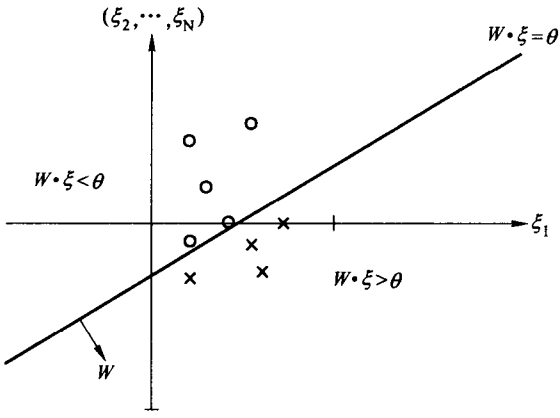

图 2.2 用线性感知器分类

图 2.3 给出另一种等价的几何解释. 定义 $x^j = O\tilde{\xi}^j$ (参见注 1.1),则线性感知器的目标成为:选取 $\widetilde{W}\in\mathbf{R}^{N+1}$,使

$$\widetilde{W}\cdot x^j>0,\quad \forall\, j=1,\cdots,J \tag{2.3}$$

如图 2.3,设 l_1,l_2 张成包含 $\{x^j\}_{j=1}^J$ 的最小扇形域,β 是其张开的角度. 于是,角度差 $\sigma=\pi-\beta$ 刻画了 $\{x^j\}_{j=1}^J$(从而 $\{\xi^j\}_{j=1}^J$)的可分性. 若 $\sigma<0$,则不可分;若 $\sigma>0$,则可分(对线性感知器,常称为**线性可分**). 并且 σ 越大,可分性越好(即 W 的允许范围越大).

容易证明,若 $\{\xi^j\}_{j=1}^J$ 线性无关,则一定是线性可分的. 在图 2.4 和图 2.5 中给出线性不可分的两个典型例子,其中图 2.4 所描绘的即为著名的 XOR 问题.

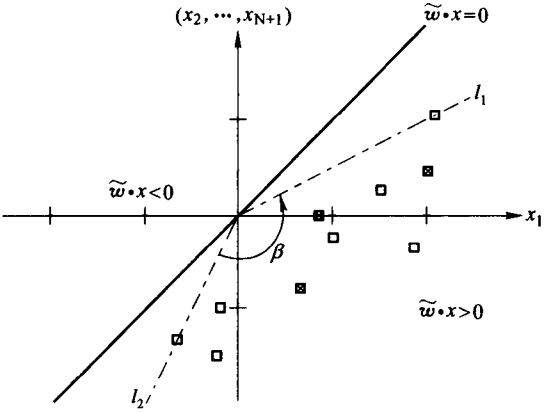

图 2.3 $\{\xi^j\}$的可分性

注:▣点是满足 $\widetilde{w}\cdot\tilde{\xi}^j<0$ 的那些样本点经过变换 $x^j=-\tilde{\xi}^j$ 得到的

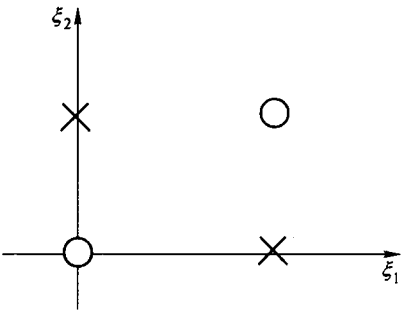

图 2.4 XOR 问题

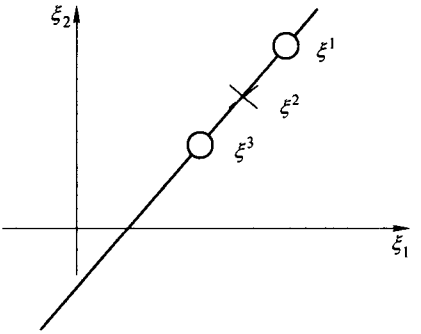

图 2.5 ξ^1,ξ^2,ξ^3 线性不可分

权向量 W 是通过学习得到的. 下面给出一种所谓**感知器学习规则**. 为简便起见,在本章其余地方,我们总假设 $\theta=0$(参见注 1.1).

输入一个样本向量 ξ^j,得到网络的当前实际输出 ζ^j,然后按下式修改当前权向量 W^{old}:

$$W^{\text{new}}=W^{\text{old}}+\Delta W \tag{2.4}$$

$$\Delta W=\frac{1}{2}(O^j-\zeta^j)\eta\xi^j \tag{2.5}$$

其中常数 $\eta>0$ 是学习速率. 如果还希望样本模式 ξ^j 别太靠近划分超平面 $\{x\mid W\cdot x=0\}$,则可以选定常数 $d\geqslant 0$,要求($x^j\equiv O^j\xi^j$)

$$W\cdot x^j\geqslant d \tag{2.6}$$

这里注意,在 W 和 x^j 长度固定的前提下,d 越大则 x^j 离划分超平面 $\{x\mid W\cdot x=0\}$ 越远. 因此,常数 d 可以理解为向量 x^j 与超平面 $\{x\mid W\cdot x=0\}$ 的距离. 这

时,式(2.5)可换成

$$\Delta W = \phi(d - W \cdot x^j)\eta x^j \tag{2.7}$$

这里 $\phi(t)$是阶梯函数:

$$\phi(t) = \begin{cases} 1, & t > 0 \\ 0, & t \leqslant 0 \end{cases}$$

注 2.2 近年来引起广泛注意的**支持向量机**神经网络的基本想法是:对给定的训练集,设法求得 d(可以是负数)的最大值 d_M,并且求得使 $W \cdot x^j = d_M$ 的那些训练样本(支持向量). 这样,就得到最优的划分平面,并且确定了对样本划分最为重要的那些支持向量,参见[ZX],[LJ].

现在,设样本集$\{\xi^j\}_{j=1}^J$按式(2.6)的意义可分,即存在$W = W^* \in \mathbf{R}^N$和 $d \geqslant 0$,使得式(2.6)对 $j = 1, \cdots, J$ 成立. 将$\{\xi^j\}$按任一顺序排成一个无穷序列$\{\xi^{(k)}\}_{k=1}^{\infty}$,使得每一 ξ^j 皆在其中出现无穷多次. 从任一初始向量 W^0 出发,依次输入 $\xi^{(1)}, \xi^{(2)}, \cdots$,按式(2.4),式(2.7)更新权值,得到权向量序列$\{W^k\}_{k=1}^{\infty}$. 我们下面来证明,迭代序列$\{W^k\}$有限步收敛,即 k 足够大后,W^k 不再改变.

收敛性证明 不失一般性,为记号简便,我们设 $W^0 = 0$, $\| W^* \| = 1$. 这样,便有

$$W^k = \eta \sum_{j=1}^{J} M(j,k) x^j \tag{2.8}$$

其中 $M(j,k)$表示在得到 W^k 的过程中,实际用到 x^j 来更新权值的次数. 这样,得到 W^k 时所有实际更新的总次数是

$$M_k = \sum_{j=1}^{J} M(j,k) \tag{2.9}$$

由于 W^*使式(2.6)成立且 $\| W^* \| = 1$,我们有

$$\begin{aligned} \| W^k \| &= \| W^k \| \, \| W^* \| \geqslant | W^k \cdot W^* | \\ &= \left| \eta \sum_{j=1}^{J} M(j,k) x^j \cdot W^* \right| \\ &\geqslant \eta \sum_{j=1}^{J} M(j,k) d = \eta M_k d \end{aligned} \tag{2.10}$$

式(2.10)给出了 $\| W^k \|$ 下界的一个估计. 接着,我们来考察 $\| W^k \|$ 的上界. 若 $W^k = W^{k-1}$(即输入向量 $\xi^{(k)}$已经被权向量 W^{k-1}正确划分),则当然有

$$\| W^k \|^2 - \| W^{k-1} \|^2 = 0 \tag{2.11}$$

反之,由式(2.4)(记 $x^{(k)} \equiv O^{(k)} \xi^{(k)}$,$O^{(k)}$是 $\xi^{(k)}$的理想输出),

$$W^k = W^{k-1} + \eta x^{(k-1)} \tag{2.12}$$

并且这时

$$W^{k-1} \cdot x^{(k-1)} < d \tag{2.13}$$

因此

$$\begin{aligned} &\| W^k \|^2 - \| W^{k-1} \|^2 \\ &= \eta^2 \| x^{(k-1)} \|^2 + 2\eta W^{k-1} \cdot x^{(k-1)} \\ &\leqslant \eta^2 D + 2\eta d \end{aligned} \tag{2.14}$$

其中 $D = \max\limits_{1 \leqslant j \leqslant J} \| x^{(j)} \|^2$. 综合式(2.11)及式(2.14),从 0 到 k 求和便得

$$\| W^k \|^2 \leqslant (\eta^2 D + 2\eta d) M_k \tag{2.15}$$

综合 $\| W^k \|$ 的下界估计式(2.10)及上界估计式(2.15)便得

$$\eta^2 d^2 M_k^2 \leqslant \| W^k \|^2 \leqslant (\eta^2 D + 2\eta d) M_k \tag{2.16}$$

因此,$\eta^2 d^2 M_k^2 \leqslant (\eta^2 D + 2\eta d) M_k$. 由此立得

$$M_k \leqslant \frac{D + 2\eta^{-1} d}{d^2} \tag{2.17}$$

收敛性得证. ■

若样本集 $\{\xi^j\}_{j=1}^J$ 不是线性可分的,则按感知器规则式(2.4),式(2.7)来求权值 W 的迭代过程不收敛. 这时,可以使用基于梯度下降法的 α-LMS(Least Mean Square)算法. 为此,对当前输入样本向量 ξ^j,定义误差函数

$$H(W) = \frac{1}{2}(O^j - W \cdot \xi^j)^2 \tag{2.18}$$

其梯度为

$$D_W H = -\epsilon^j \xi^j, \quad \epsilon^j \equiv O^j - W \cdot \xi^j \tag{2.19}$$

为使 $H(W)$ 减小,W 应朝 $H(W)$ 的梯度反方向走,因此迭代公式为

$$\begin{aligned} &W^{\text{new}} = W^{\text{old}} + \eta \epsilon_{\text{old}}^j \xi^j \\ &\epsilon_{\text{old}}^j \equiv O^j - W^{\text{old}} \cdot \xi^j \end{aligned} \tag{2.20}$$

容易推得

$$\begin{aligned} \epsilon_{\text{new}}^j &\equiv O^j - W^{\text{new}} \cdot \xi^j \\ &= O^j - W^{\text{old}} \cdot \xi^j - \eta \epsilon_{\text{old}}^j \| \xi^j \|^2 \\ &= \epsilon_{\text{old}}^j (1 - \eta \| \xi^j \|^2) \end{aligned}$$

从而,为使 W^{new} 满足 $|\epsilon_{\text{new}}^j| < |\epsilon_{\text{old}}^j|$,应有

$$|1 - \eta \| \xi^j \|^2| < 1$$

或等价地

$$0 < \eta \| \xi^j \|^2 < 2 \tag{2.21}$$

总之,我们应有

$$W^{\text{new}} = W^{\text{old}} + \frac{\alpha \epsilon^j \xi^j}{\| \xi^j \|^2} \tag{2.22}$$

其中

$$0 < \alpha < 2 \tag{2.23}$$

此即为 α-LMS 算法. 实际应用中,α 常选为 $0.1 < \alpha < 1$.

迭代过程式(2.22)可以在线性输出 $W \cdot \xi^j$ 都有(或大多数有)正确的符号后停止.另外,也可以直接将线性输出作为网络输出,这时的网络相当于一个线性逼近器.

注 2.3 迭代公式(2.22)是一种 δ-学习算法. 式(2.23)是 δ-学习算法收敛的一个典型的必要条件.

注 2.4 用 α-LMS 算法来迭代确定权值 W 时,样本向量 ξ^j 可以按 $j=1,2,\cdots,N,1,2,\cdots,N,\cdots$ 顺序依次输入,也可以按随机的顺序输入. 本章以后各节的类似问题,都可以照此办理.

注 2.5 网络结构、工作流程和学习方法,是一个神经网络的三大要素. 对于线性感知器来说,这三大要素分别由图 1.1($g(x)\equiv\mathrm{sgn}(x)$),式(2.2)和式(2.4)给出.

§1.3 Madaline 网络

单个的自适应线性感知器常简称为 Adaline(由 Adaptive 和 linear 二词合成). 它的分类能力是很有限的. 可以将多个 Adaline 组合起来,得到 **Madaline (Multiple Adaline) 网络**. 本节我们介绍两种典型的 Madaline 网络, 即MR Ⅰ 和 MR Ⅱ (Madaline Rule Ⅰ, Ⅱ).

MR Ⅰ 网络

一种典型 **MR Ⅰ 网络**的连接方式如图 3.1 所示.

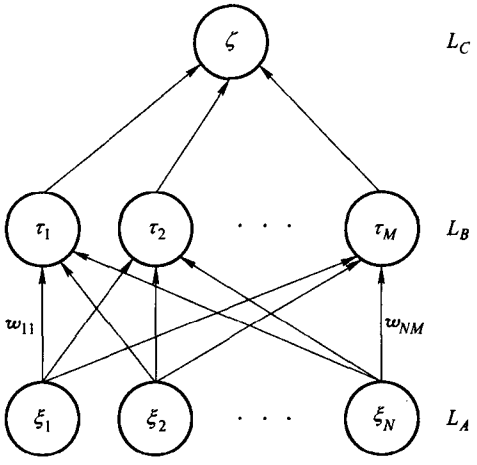

图 3.1 MR Ⅰ 网络结构

通过学习选定权值矩阵 W 后，MR Ⅰ网络的工作方式如下：

$$\tau_m = \mathrm{sgn}\Big(\sum_{n=1}^{N} W_{nm}\xi_n\Big), \quad m = 1, \cdots, M \tag{3.1a}$$

$$\zeta = \mathrm{sgn}\Big(\sum_{m=1}^{M} \tau_m\Big) \tag{3.1b}$$

外界输入信息 $\xi=(\xi_1, \cdots, \xi_N)^{\mathrm{T}}$ 先通过输入层 L_A，经 L_B 层的 M 个 Adaline 感知器处理，然后再由 L_C 层的多数表决器给出最后输出结果 ζ. L_C 层的多数表决器功能是给定的，无须学习. L_C 层也可以由其他各种与门、非门等组成. L_B 层的权值矩阵 W 通过学习得到. 初始权值 W^0_{nm} 可随机地取接近于零的数. 下面介绍权值 W 的一种学习方法，即 **MR Ⅰ 规则**.

输入一个样本向量 ξ^j，如果最终输出的 ζ^j 与相应理想输出 O^j 一致，则不加调整；否则，对当前权值 W 作调整. 例如，设 $M=5, O^j=1, \zeta^j=-1$. 这时，若有三个(或四个、五个)Adaline 元件 τ_m 输出 -1，则选定其中一个(或相应地二个，三个)元件来修改其权值. 选定的标准是“最小扰动原则”，即在给出错误输出的那些 Adaline 元件中，选取其线性输出 $h_m = \sum_{n=1}^{N} W_{nm}\xi_n$ 最接近于零的那一个(或几个). 每次调整某 Adaline 元件权值时，可以沿 LMS 方向(参见式(2.20))将相应权值向量移动得足够远，使得该元件输出反号，也可以只按 α-LMS 算法来作微调. 对输入样本 ξ^j 做完如上调整后，再输入另一个样本重复以上过程.

MR Ⅱ 网络

MR Ⅱ 网络是一种多层前传网络，有一个输入层、一个输出层和若干隐层. 除输出层外，每一层由若干个 Adaline 组成. 任一 Adaline 与其下一层中每一个 Adaline 通过权值相连. 选定权值 W 以后，对给定的输入 $\xi=(\xi_1, \cdots, \xi_N)^{\mathrm{T}}$，输出 ζ 由下式给出(以一个隐层、一个输出单元为例)：

$$\tau_m = \mathrm{sgn}\Big(\sum_{n=1}^{N} w_{nm}\xi_n\Big), \quad m = 1, \cdots, M \tag{3.2a}$$

$$\zeta = \mathrm{sgn}\Big(\sum_{m=1}^{M} W_m\tau_m\Big) \tag{3.2b}$$

给定一组输入输出样本模式 $\{\xi^j, O^j\}_{j=1}^{J}, O^j = \pm 1$. 权值学习的任务(即 w 和 W 的选择)是从任一随机给定的接近于零的权值出发，依次输入样本模式，逐步调整权值，使得对所有(或绝大部分)样本输入 ξ^j，都能得到理想输出 O^j.

针对 MR Ⅱ 网络的 **MR Ⅱ 算法**如下：输入一个样本模式 ξ^j，若网络实际输出与理想输出不符，则调整各层权值. 先调整第一隐层中线性输出最接近于零的那个 Adaline 与输入层的连接权值(例如用 α-LMS 方法)，使其输出变号，这时以下各层输出均有变化. 若最终的线性输出误差有所减少(例如理想输出是 $+1$，而最

终线性输出由 -0.5 变为 -0.1),则接受刚修改过的那个 Adaline 的新权值;否则恢复原权值. 然后,再调整第一层中其余 Adaline 中线性输出最接近于零的那一个. 第一层中所有 Adaline 都如此处理一遍以后,再按这样的"最小扰动原则",选定两个一组,三个一组,……使其输出同时变号,并按最终线性输出误差是否减少来决定是否接受相应权值的更新. 处理完第一层后,再依次处理以后各层,直到最后一层. 如有必要,可回头再从第一层开始. 每次输入一个 ξ^j,都重复以上过程,直至得到理想输出 O^j. 所有 ξ^j 都训练完一遍以后,还需再重复输入 $\{\xi^j\}_{j=1}^J$,直到所有(或大部分)ξ^j 都能无须调整权值而得到理想输出 O^j 为止.

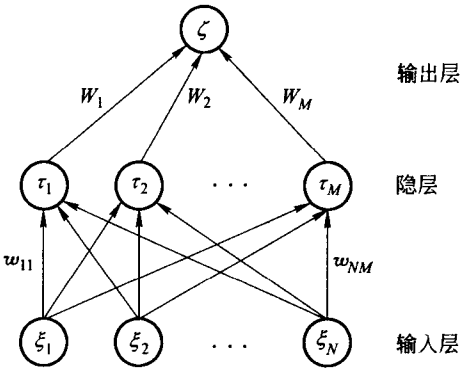

图 3.2 MRⅡ网络结构(一个隐层、一个输出单元)

对许多问题的测试表明 MRⅠ、MRⅡ算法是收敛的,但数学上并无严格证明.

§1.4 BP 网 络

BP 网络是现在应用最为广泛的神经网络. 它采用光滑活化函数,具有一个或多个隐层,相邻两层之间通过权值全连接. 它是前传网络,即所处理的信息逐层向前流动. 而当学习权值时,却是根据理想输出与实际输出的误差,由前向后逐层修改权值(误差的后向传播, 即 Back Propagation).

BP 网络拓扑结构与图 3.2 完全相同,只是用光滑活化函数取代了符号函数,见图 4.1(以带一个隐层和一个输出单元的 BP 网络为例).

选定一个非线性光滑活化函数 $g:\mathbf{R}^1\to\mathbf{R}^1$,并按稍后给出的规则确定了权矩阵 $W=\{W_{mp}\}_{1\leqslant m\leqslant M,1\leqslant p\leqslant P}$ 和 $w=\{w_{pn}\}_{1\leqslant p\leqslant P,1\leqslant n\leqslant N}$ 之后,对任一输入信息向量 $\xi=(\xi_1,\cdots,\xi_N)^{\mathrm{T}}\in\mathbf{R}^N$,网络的实际输出为

$$\zeta_m = g(W_m\cdot\tau)=g\Big(\sum_{p=1}^{P}W_{mp}\tau_p\Big),\quad m=1,\cdots,M \tag{4.1a}$$

其中隐层输出为

$$\tau_p = g(w_p \cdot \xi) = g\Big(\sum_{n=1}^{N} w_{pn}\xi_n\Big), \quad p = 1,\cdots,P \tag{4.1b}$$

现在,假设给定一组样本输入向量$\{\xi^j\}_{j=1}^J \subset \mathbf{R}^N$及相应的理想输出$\{O^j\}_{j=1}^J \subset \mathbf{R}^M$,并记$\{\zeta^j\}_{j=1}^J \subset \mathbf{R}^M$为相应的网络实际输出. 定义误差函数

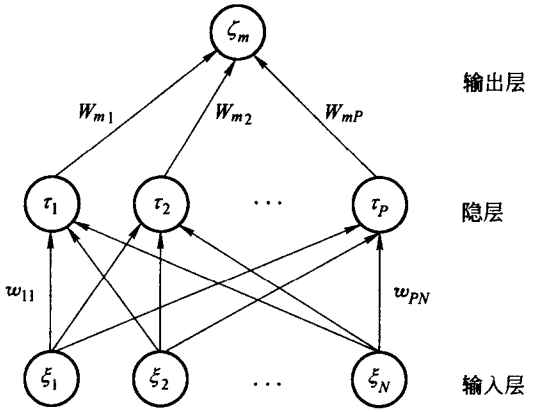

图 4.1 BP 网络结构

$$E(W,w) \equiv \frac{1}{2}\sum_{j=1}^{J} \| O^j - \zeta^j \|^2 = \frac{1}{2}\sum_{j=1}^{J}\sum_{m=1}^{M}\Big[O_m^j - g\Big(\sum_{p=1}^{P} W_{mp}\, g\Big(\sum_{n=1}^{N} w_{pn}\xi_n^j\Big)\Big)\Big]^2 \tag{4.2}$$

权值矩阵 W 和 w 的确定(即学习过程)应使误差函数 $E(W,w)$达到极小. 为此,一个简单而又常用的方法是梯度下降法. 取当前权值 W_{mp}的改变量为

$$\begin{aligned} \Delta W_{mp} &= -\eta \frac{\partial E}{\partial W_{mp}} \\ &= \eta \sum_{j=1}^{J} (O_m^j - \zeta_m^j) g'(H_m^j) \tau_p^j \\ &= \eta \sum_{j=1}^{J} \Delta_m^j \tau_p^j \end{aligned} \tag{4.3}$$

其中 $\eta > 0$ 为学习速率,

$$\Delta_m^j = (O_m^j - \zeta_m^j) g'(H_m^j) \tag{4.4}$$

而

$$H_m^j = \sum_{p=1}^{P} W_{mp}\tau_p^j \tag{4.5}$$

是隐层单元对第 m 个输出层单元的线性输入. 进一步,我们可以得到当前权值 w_{pn}的改变量为: